algorithms

MDPI

Article

Enhancing Semi-Supervised Learning in Educational Data Mining Through Synthetic Data Generation Using Tabular Variational Autoencoder

Georgios Kostopoulos [1,*], Nikos Fazakis [1], Sotiris Kotsiantis [1] and Yiannis Dimakopoulos [2]

[1] Department of Mathematics, University of Patras, 26504 Rion, Greece; fazakis@ece.upatras.gr (N.F.); sotos@math.upatras.gr (S.K.)
[2] Department of Chemical Engineering, University of Patras, 26504 Rion, Greece; dimako@chemeng.upatras.gr
* Correspondence: kostg@sch.gr

Abstract

This paper presents TVAE-SSL, a novel semi-supervised learning (SSL) paradigm that involves Tabular Variational Autoencoder (TVAE)-sampled synthetic data injection into the training process to enhance model performance under low-label data conditions in Educational Data Mining tasks. The algorithm begins with training a TVAE on the given labeled data to generate imitative synthetic samples of the underlying data distribution. These synthesized samples are treated as additional unlabeled data and combined with the original unlabeled ones in order to form an augmented training pool. A standard SSL algorithm (e.g., Self-Training) is trained using a base classifier (e.g., Random Forest) on the combined dataset. By expanding the pool of unlabeled samples with realistic synthetic data, TVAE-SSL improves training sample quantity and diversity without introducing label noise. Large-scale experiments on a variety of datasets demonstrate that TVAE-SSL can outperform baseline supervised models in the full labeled dataset in terms of accuracy, F1-score and fairness metrics. Our results demonstrate the capacity of generative augmentation to enhance the effectiveness of semi-supervised learning for tabular data.

Keywords: Educational Data Mining; fairness; prediction; semi-supervised learning; synthetic data; variational auto-encoder

Academic Editors: Antonio Sarasa Cabezuelo and María Estefanía Avilés Mariño

Received: 21 September 2025
Revised: 13 October 2025
Accepted: 16 October 2025
Published: 19 October 2025

Citation: Kostopoulos, G.; Fazakis, N.; Kotsiantis, S.; Dimakopoulos, Y. Enhancing Semi-Supervised Learning in Educational Data Mining Through Synthetic Data Generation Using Tabular Variational AutoEncoder. *Algorithms* **2025**, *18*, 663. https://doi.org/10.3390/a18100663

1. Introduction

One of the most significant advancements in the field of education over the past two decades has been the integration of emerging technologies. Today, Information and Communication Technologies play a vital role in the educational process, providing both educators and learners with a wide range of interactive learning environments that support instruction and promote educational quality [1]. Consequently, substantial volumes of data are continuously generated and stored in institutional databases and information systems, reflecting students' learning behaviors, online activities, and academic achievements.

The growing need to analyze various types of educational data and extract meaningful insights has led to the emergence of Educational Data Mining (EDM), which is recognized as a rapidly evolving research domain [2]. EDM primarily focuses on the development and application of data mining techniques to educational data derived from diverse learning environments [3]. Its overarching goal is to address key educational challenges, ultimately aiming to enhance the learning process and improve the overall quality of education [4]. One of the most widely studied applications of EDM is prediction, which involves building

a machine learning (ML) model by training a supervised algorithm on a labeled dataset, and subsequently applying it to predict unknown or future outcomes of students [5].

Unfortunately, in many practical scenarios, obtaining a sufficient amount of labeled training data is difficult due to the high cost of data labeling. This challenge has led to the emergence of novel machine learning approaches such as Semi-Supervised Learning (SSL) [6]. SSL represents a core branch of Weakly Supervised Learning, aiming to leverage a small set of labeled examples in combination with a large set of unlabeled ones to build highly accurate and robust predictive ML models [7]. A variety of methodologies grounded in the core principles of SSL have been developed and successfully applied across a wide range of domains, including web mining, text mining, image processing, information retrieval, and bioinformatics [8].

In recent years, a growing body of research has explored the effectiveness of SSL methods in the field of education, primarily for predictive tasks. Consequently, numerous studies have reported highly promising results, often outperforming traditional supervised learning approaches [9]. SSL algorithms enable the development of highly accurate early-warning models for the timely identification of students at risk of academic failure. This facilitates the implementation of targeted support measures and specialized intervention strategies aimed at enhancing the learning outcomes of underperforming students [10].

While educational data offer immense research and pedagogical potential, acquiring them is frequently hampered by high costs and time-consuming procedures. Moreover, stringent privacy regulations impose legal barriers that limit access to raw student records, complicate cross-institutional sharing, and require strict oversight of personally identifiable information. Furthermore, existing ethical frameworks exhibit concerns and gaps that limit their effectiveness in guiding the ethical development and implementation of educational data initiatives [11]. These privacy and legal considerations, when combined with limited technical expertise and budget constraints, result in under-collected, biased, and/or incomplete educational datasets that decrease their utility for robust machine learning.

Synthetic data generation and augmentation methods have emerged as practical alternatives to compensate for limited real-world educational data. Unlike expensive and laborious data collection processes, synthetic data can be generated easily and rapidly at scale while maintaining statistical fidelity to the original distributions, enabling researchers to explore varied hypothetical scenarios, balance underrepresented student groups, and simulate events that are rarely captured in real datasets.

The primary contribution of this research is TVAE-SSL, a novel framework that combines generative modeling and SSL to improve predictive performance and fairness under low-label settings. Unlike traditional SSL methods that are solely based on accessible unlabeled data, TVAE-SSL intentionally incorporates high-quality synthetic samples drawn from the limited labeled set, generated by a Tabular Variational Autoencoder (TVAE), into the unlabeled set. This approach enriches the data with diversity without adding label noise and enables stronger learning. The framework is also modular and independent of SSL algorithm selection and base classifier choice. We show via large-scale experiments that TVAE-SSL not only boosts typical performance measures such as accuracy and F1-score but also enhances fairness measures such as demographic parity and equalized odds—demonstrating its ability to build fair and resilient models from small labeled datasets.

The rest of the paper is organized as follows: Section 2 provides an overview of recent applications of SSL methods within the educational domain, with particular emphasis on synthetic data generation techniques and fairness considerations in the field of EDM. Section 3 defines the research methodology, while Section 4 describes the experimental framework. Section 5 presents the experimental results, while Section 6 provides a detailed discussion of these results, emphasizing the key findings and their implications. Finally, Section 7 concludes the research outcomes and suggests future research directions.

2. Related Work

A plethora of studies have been conducted to address prediction tasks within the field of EDM. These tasks primarily involve predicting student performance, dropout, grade level, and final grade [12]. Accurate predictions of student outcomes can enable educators and tutors to offer timely interventions for underperforming students, thereby reducing the risk of academic failure and enhancing the overall quality of education [10].

2.1. Overview of Semi-Supervised Learning in EDM

Student performance prediction is a central focus in the field of EDM. Typically, this involves two main aspects: predicting whether a student will pass or fail a unit or course, and classifying students into multiple categories based on their grade level.

The effectiveness of various SSL algorithms was investigated in [13] for predicting student performance in the final examination of a one-year distance learning course. Several SSL algorithms were applied using four classifiers as base learners: the Naïve Bayes (NB) classifier, the C4.5 decision tree, the k-nearest neighbors (k-NN) algorithm, and the Sequential Minimal Optimization (SMO) algorithm. The results demonstrated that SSL algorithms were notably effective in the early prediction of low-performing students. Among the methods evaluated, Tri-Training using three C4.5 decision trees as base classifiers, each configured with different parameters to ensure diversity, outperformed not only the other SSL algorithms but also the supervised C4.5 classifier. A co-training method was developed in [9] for the early prediction of student performance in distance higher education, leveraging two distinct and independent feature views: academic achievement data and activity in the Learning Management System. A plethora of experiments was conducted to evaluate the effectiveness of the proposed method in comparison with various Self-Training and co-training variants across three scenarios, each based on a different ratio of labeled data within the training set: 2.5%, 10%, and 15%. The proposed method consistently outperformed all other SSL algorithms as well as several supervised classification algorithms in terms of accuracy and F1-score, regardless of the labeled ratio and the base learner used.

Various SSL algorithms were applied in [14] to predict the grade level (Poor, Good, Very Good, Excellent) of high school students in the final examinations of the mathematics module at the end of the academic year. The experimental results demonstrated that Self-Training, Tri-Training, and co-training, utilizing the Naïve Bayes (NB) algorithm as the base classifier, achieved superior performance, with accuracy ranging from 64.41% to 67.35% by the midpoint of the academic year.

Student dropout remains a significant concern in education, particularly in distance learning environments such as MOOCs. The effectiveness of the SSL approach was investigated in [15] for predicting students at risk of dropping out in a distance higher education course. A series of experiments were conducted, dividing the academic year into three consecutive time periods, and employing various semi-supervised classification algorithms. The experimental results indicated that Tri-Training and Self-Training achieved superior performance, with accuracy ranging from 71.74% to 76.73% prior to the midpoint of the academic year, thereby supporting the implementation of timely intervention measures. A multi-view SSL model based on behavioral features of students was introduced in [16] for predicting student dropout in a MOOC. To this end, data reflecting different types of learning behavior of students were collected and used to form multiple views of behavioral features. Experiments on the KDD Cup 2015 dataset demonstrated that the proposed method outperformed established supervised methods.

Grade prediction represents another important task in EDM. This task is typically approached using regression algorithms, as the target variable is continuous. An ensemble-based algorithm, termed the Multi-Scheme Semi-Supervised Regression (SSR) approach,

was introduced in [17] for predicting undergraduate students' final examination grades in a one-year distance learning course. Three k-NN regressors were employed within a Self-Training framework to iteratively expand the labeled dataset by leveraging unlabeled data. Subsequently, a Random Forest (RF) regressor was utilized to build the regression model. The proposed method outperformed conventional regression methods in terms of four metrics: Mean Absolute Error, Relative Absolute Error, Root Mean Squared Error and Pearson Correlation Coefficient. A multi-view SSR algorithm was implemented in [18] to predict the final examination grades of undergraduate students enrolled in a one-year distance learning course. The prediction was conducted at two distinct time points prior to the midpoint of the academic year. Additionally, the study investigated the influence of input attributes on the target one, generating a range of interpretable visualizations that illustrate their impact on the output of the ML model. The experimental results revealed that the highest-performing students were those who achieved high grades in the two compulsory written assignments completed during the first semester.

A comprehensive review of the applications of SSL in the fields of EDM and Learning Analytics is presented in [10].

2.2. Synthetic Data Generation

The quality of training data significantly influences the efficiency of ML classification models. Poor data quality leads to models with low accuracy, which may result in incorrect predictions. Moreover, in many practical scenarios, obtaining a sufficient amount of labeled training data is difficult due to the high cost of data labeling. Furthermore, the dissemination of data is frequently constrained by privacy and fairness considerations across many domains, such as education [19]. In light of these concerns, synthetic data generation emerges as a compelling alternative, facilitating secure data sharing and application beyond the limitations of real-world datasets.

Concerning the educational field, synthetic data offers a plethora of advantages, such as the following:

(1) Enhanced Privacy Protection
Synthetic data eliminates the risk of exposing sensitive student information, ensuring compliance with data protection regulations [20].
(2) Mitigation of Data Scarcity
Many educational datasets suffer from limited sample sizes. Synthetic data can augment datasets, improving model generalizability [21].
(3) Bias reduction and Fairness Improvement
Real-world educational data often reflects systemic biases. Synthetic data can be strategically generated to balance underrepresented classes, leading to fairer predictive models [22].
(4) Controlled Experimentation and Scenario Testing
Researchers can simulate hypothetical educational scenarios to test predictive models without needing real-world trials [23].
(5) Accelerated Research and Open Collaboration
Synthetic datasets can be shared freely among researchers, enabling reproducibility and collaboration without legal or ethical restrictions [24].
(6) Robustness against Overfitting
By introducing controlled variations, synthetic data can help train models that generalize better to unseen real-world data, reducing overfitting risks [25].
(7) Cost and Time Efficiency
Collecting real educational data is often time-consuming and expensive. Synthetic data generation provides a scalable alternative [26].

2.3. Fairness Considerations in EDM

ML fairness is an emerging area that studies how to ensure that the outputs of a model do not depend on sensitive attributes in a way that is considered unfair. For example, in a model that predicts student performance based on previous school records, this could mean ensuring that the decisions do not depend on gender. However, as EDM systems become more integrated into decision-making processes, concerns regarding fairness and equity have intensified. Fairness in EDM refers to the absence of bias and discrimination in data collection, algorithmic processing, and interpretation, particularly concerning sensitive attributes such as race, gender, socio-economic status, and disability [27].

A major challenge in ensuring fairness arises from historical biases embedded in educational datasets. These biases can be inadvertently perpetuated or amplified by ML models, leading to unfair outcomes such as unequal access to resources or incorrect at-risk predictions [28]. For example, predictive models trained on imbalanced data may overidentify students from marginalized groups as underperforming, triggering disproportionate interventions or stigmatization. Hence, rigorous bias detection and mitigation strategies, such as fairness-aware learning algorithms or preprocessing techniques, are essential in EDM [22]. Furthermore, fairness is context-dependent and multidimensional. It is not only a technical issue but also an ethical and social one. For instance, ensuring fairness might involve balancing multiple trade-offs between accuracy and equity or between group fairness and individual fairness [29]. Stakeholder engagement, including input from educators, students, and policymakers, is critical to defining what constitutes fairness in a given educational context and to guiding the design of just and inclusive systems.

While prior work has extensively explored SSL methods and fairness-aware machine learning separately, few prior studies exist that thoroughly consider the combination of generative data augmentation with SSL for addressing accuracy and fairness simultaneously in low-label scenarios. Most SSL methods were assumed to receive relatively balanced or clean labeled datasets as input and therefore limited their robustness when they actually encountered real-world scenarios where the labeled datasets are scarce and possibly biased. Similarly, generative models have been applied to synthetic data generation but with performance improvement as the primary focus and fairness as a secondary concern. The novel TVAE-SSL model bridges the gap by leveraging Tabular Variational Autoencoders (TVAEs) to generate realistic synthetic samples whose adoption not only improves classification performance but also encompasses a method of fairness control and measurement. By combining these two research streams, our solution offers a different perspective regarding how synthetic data generation can help SSL address both predictive accuracy and fairness, thereby addressing an essential gap in the existing literature.

3. Research Methodology

This section provides a detailed description of the datasets used, the synthetic data generation process, and the SSL algorithms employed, along with the performance and fairness metrics considered.

3.1. Datasets

The experimental procedure utilizes four (4) classification datasets from the field of EDM. The key characteristics of these datasets are summarized in Table 1, including the number of instances, the number of attributes, the sensitive attributes, the number of output classes, and the corresponding prediction task.

Table 1. Dataset structure.

Dataset	# Instances	# Attributes	Sensitive Attributes	# Output Classes	Prediction Task
student	344	12	gender	2	dropout
eap2024	445	6	gender	2	success
cert	936	33	employment status	2	certification
MBA	6194	9	gender	3	admission

In the datasets under consideration, the majority employ gender as the sensitive attribute, with the exception of the cert dataset, where the sensitive feature is employment status. This distinction highlights the relevance of different demographic and socio-economic factors in fairness-aware machine learning applications. In the case of gender, the female group is regarded as the protected class, reflecting the need to ensure that predictive outcomes do not disadvantage women in educational or admission-related tasks. Conversely, for the cert dataset, individuals categorized as not employed constitute the protected group, since their employment status could introduce bias in certification predictions. The analysis ensures that fairness interventions can be appropriately targeted to mitigate discrimination in each predictive setting.

The first two datasets pertain to students enrolled in the twelve-course "Computer Science" module offered by the Hellenic Open University. Each course module, such as "Introduction to Informatics", requires the submission of four written assignments, attendance at five optional contact sessions, and participation in a final examination. To successfully complete the course, students are required to submit a minimum of three written assignments with an average grade of at least five on a ten-point scale, and to achieve a minimum score of five in the final examination.

The first dataset comprises 344 instances described by twelve attributes (Table 2). The first seven attributes pertain to students' demographic characteristics and general employment information, namely gender, age, marital status, number of children, employment status, computer knowledge, and job correlation with computer skill requirements. Several studies have demonstrated that these factors significantly influence student success in distance learning environments [30,31]. The next four attributes capture students' performance on the first two written assignments WA_1, WA_2 and their attendance in the first two optional contact sessions with their tutor OCS_1, OCS_2. These attributes become available progressively throughout the academic year. Attendance at contact sessions is recorded as binary values, with 1 indicating presence and 0 indicating absence. Assignment grades range from -1 to 10, where -1 indicates non-submission. The target attribute indicates whether a student dropped out of the course.

Table 2. Description of the 1st dataset.

Attribute	Type	Values	Description
Gender	binary	{male, female}	Student gender
Age	binary	{<32, >=32}	Student age
Marital status	binary	{single, married}	Student marital status
Children	integer	[0, 4]	Number of children
Employment status	nominal	{unemployment, part-time, full-time}	Employment status
Computer knowledge	binary	{yes, no}	Computer knowledge
Computer-skilled job	ordinal	{low, medium, high}	Job correlation with computer skill requirements
OCS_1, OCS_2	binary	{0, 1}	Absence/Presence in each OCS
WA_1, WA_2	real	$[-1, 10]$	Grade of each WA
dropout	binary	{yes, no}	Output class

The second dataset comprises 445 instances described by six attributes (Table 3). The first attribute corresponds to the students' gender, while the next four attributes capture students' performance on the first four written assignments WA_1, WA_2, WA_3, WA_4. Assignment grades range from -1 to 10, where -1 indicates non-submission. The target attribute indicates course completion status as either pass or fail.

Table 3. Description of the 2nd dataset.

Attribute	Type	Values	Description
Gender	binary	{male, female}	Student gender
WA_1, WA_2, WA_3, WA_4	real	$[-1, 10]$	Grade of each WA
Success	binary	{pass, fail}	Output class

The third dataset comprises data (Table 4) from a 11-week MOOC course in the context of the Erasmus+ Sector Skills Alliance project called "DevOps Competences for Smart Cities" [32]. The course was organized into 15 modules, with 1–2 modules released weekly. Each module contained 2–5 discrete learning units, and every learning unit concluded with an automatically graded multiple-choice quiz for assessment. The dataset consists of eleven attributes capturing various aspects of students' personal information, ten attributes about students' grades in quizzes (100-point scale) during the first two weeks of the course and overall grades in modules 1 and 2, and twelve numerical attributes concerning students' activity in the online learning platform, such as number of views, posts, discussions and connections, as well as the total time devoted to the first two modules of the course. The target attribute specifies whether a student successfully obtained a certificate upon completion of the course.

Table 4. Description of the 3rd dataset.

Attribute	Type	Values	Description
Mother tongue	nominal		Mother tongue
Education level	nominal		Education level
Employment status	nominal		Employment status
Current occupation	nominal		Current occupation
Occupation experience	real		Occupation experience in years
Work time	real		Average working hours per week
Technical skills	nominal		Technical English language skills
Digital skills	nominal		Digital proficiency skills
MOOC experience	binary	{yes, no}	Previous experience in MOOCs
Study week hrs	real		Average study hours per week
W1 Ui Grade	real	[0, 100]	Week 1 Unit i Assessment Quiz Grade, i $\in \{1,2,3,4,5\}$
W2 Ui Grade	real	[0, 100]	Week 2 Unit i Assessment Quiz Grade, i $\in \{1,2,3\}$
Mod i Grade	real	[0, 100]	Overall grade in Module i, i $\in \{1,2\}$
Views Intro	real		Number of views in introductory forum
Views Anno	real		Number of views in announcements forum
Views Mod i	real		Number of forum views in Module i, i $\in \{1,2\}$
Posts Mod i	real		Number of posts in Module i, i $\in \{1,2\}$
Discu Mod i	real		Number of discussions in Module i, i $\in \{1,2\}$
Conne Mod i	real		Average connections per day in Module i, i$\in \{1,2\}$
Mod i Time	real		Total time dedicated in Module i (mins), i $\in \{1,2\}$
Certificate	binary	{yes, no}	Output class

The fourth dataset was sourced from the Kaggle platform and pertains to students enrolled in the Wharton School's Class of 2025 statistics course. It comprises 6194 instances described by nine attributes (Table 5). Six attributes represent the demographic charac-

teristics of the applicants. In addition, one attribute captures the applicant's Grade Point Average (GPA), and another attribute records their Graduate Management Admission Test (GMAT) score. The target attribute indicates the outcome of the admission process.

Table 5. Description of the 4th dataset.

Attribute	Type	Values	Description
Gender	binary	{male, female}	Applicant's gender
International	binary	{true, false}	International student
Major	nominal	{Business, STEM, Humanities}	Undergraduate major
Race	nominal		Racial background of the applicant
Work industry	nominal		Industry of the applicant's previous work experience
Work experience	integer		Work experience in years
GPA	real	[0, 4]	Grade Point Average of the applicant
GMAT	integer	[0, 800]	GMAT score of the applicant
Admission	nominal	{admit, wait list, deny}	Output class

3.2. SSL Methods

The experimental process involved the application of several self-labeling methods proposed in the literature, which have been successfully employed in the educational domain to address a variety of prediction tasks [10].

Self-Training [33] is an iterative method used to assign labels to unlabeled data and is commonly regarded as a reference approach in the field of SSL. According to this method, a base classifier is initially trained on a small set of labeled examples (L). The trained model is then used to predict labels for the unlabeled instances and the most confidently predicted samples are added to the labeled set for subsequent retraining. This process is repeated iteratively until a convergence criterion is met or no further improvement is observed.

SetRed (Self-Training with Editing) is an SSL algorithm designed to improve the robustness of Self-Training by incorporating a noise-reduction mechanism that filters out unreliable pseudo-labels [34]. Unlike traditional Self-Training methods, which risk reinforcing errors by adding misclassified unlabeled instances to the training set, SetRed introduces an editing step to selectively remove mislabeled examples before retraining. This editing phase typically uses techniques such as k-nearest neighbors (k-NN) to assess the consistency of each pseudo-labeled instance with its local neighborhood, retaining only those that align with the majority of their neighbors. By doing so, SetRed reduces the propagation of labeling noise, leading to more accurate and stable classifiers. This makes it particularly useful in real-world applications like EDM, where labeled data may be limited and class distributions are often imbalanced or noisy.

Co-training is based on the assumption that each example in a dataset can be represented by two distinct and conditionally independent sets of attributes, referred to as views [35]. Two classifiers are trained separately on each view using a small set of labeled examples (L) and the most confident predictions of each classifier on the set of unlabeled examples (U) augment the training set of the other until some stopping criterion is met. Co-training is a characteristic paradigm of disagreement-based SSL methods [36].

Co-Forest (Co-training Random Forest) is a semi-supervised ensemble algorithm that extends the traditional Random Forest by incorporating the co-training paradigm to leverage both labeled and unlabeled data for improved classification performance [37]. Co-Forest trains multiple decision tree learners on different data subsets, allowing each learner to iteratively label and augment the training set of the others with high-confidence predictions from unlabeled instances. This collaboration enhances generalization, especially when labeled data are scarce, a common scenario in EDM. The algorithm preserves the

diversity and robustness benefits of RF while exploiting the complementary information present in multiple learners to reduce error rates.

Co-training by Committee (CoBC) is another SSL algorithm that combines the principles of co-training and ensemble learning to improve classification performance using both labeled and unlabeled data [38]. CoBC employs a committee of diverse classifiers trained on randomly sampled subsets of the feature space and labeled data. During the iterative training process, each classifier predicts labels for a portion of the unlabeled data, and only high-confidence predictions that achieve consensus among the committee members are added to the labeled set. This collaborative strategy minimizes the propagation of errors common in self-labeling methods while enhancing model robustness through ensemble diversity.

Two multi-view methods that leverage the advantages of random subspace techniques and ensemble learning are Rasco and Rel-Rasco. Rasco (Random Subspace for Co-training) [39] is an extension of the co-training algorithm that operates on multiple randomly generated subspaces of the features. These subspaces can be interpreted as distinct views of the feature space. Separate classifiers are each trained on a small set of labeled examples within each subspace, and their predictions on an unlabeled instance are aggregated to determine its label. In this manner, the classifiers complement one another by identifying different patterns within the dataset, based on the assumption that they are typically sensitive to distinct subsets of features. The labeled subsets are incrementally augmented, and the classifiers are re-trained through an iterative learning process. Rel-Rasco (Relevant Random Subspace Co-training) [40] is an improved variant of Rasco. Instead of relying on random subspace generation, this approach systematically constructs feature subspaces by leveraging relevance scores, computed through the mutual information between each feature and the class labels.

Tri-Training is a widely used single-view multiple-classifier SSL method [41]. It employs three classifiers in an iterative learning process to label the unlabeled data. Specifically, if the two classifiers agree on the label of an unlabeled example, that example is then labeled by the third classifier and incorporated into the labeled dataset (L).

In addition, three widely recognized ensemble algorithms, namely Random Forests (RFs), Extreme Gradient Boosting (XGB) and Histogram-based Gradient Boosting (HGB), were chosen as base classifiers due to their established effectiveness and frequent use in SSL research studies:

(1) RF is a well-known bagging tree-based method [42]. It relies on bootstrap aggregation (bagging) to construct random ensembles of decision trees, while a voting strategy is employed to predict the class label for new data.
(2) XGB [43] is a highly efficient and fast-to-execute classification algorithm, a representative paradigm of gradient tree boosting, commonly referred to as gradient boosting.
(3) HGB [44] is another high-performance gradient boosting algorithm. In contrast to RF, where tree models are trained in parallel, HGB builds and adds trees sequentially by discretizing continuous feature values to binary for constructing feature histograms during the training phase.

3.3. Performance Measures

The performance of the SSL methods was evaluated in terms of accuracy and F1-score results. Accuracy measures the effectiveness of a classifier in correctly predicting the label of a previously unseen instance. The F1-score is a widely adopted metric for evaluating the performance of binary and multiclass classifiers, especially in scenarios involving imbalanced datasets [44]. It is the harmonic mean of precision (p) and recall (r).

In addition, two fairness metrics, which are commonly recognized as prevalent threshold-dependent measures in the relevant literature, were employed. Demographic

Parity Difference (DPdiff) is a group fairness metric that quantifies the degree to which the outcomes of a predictive model are independent of a sensitive attribute, such as gender, race, or socio-economic status. Formally, it measures the absolute difference in the probability of receiving a positive outcome between groups defined by the sensitive attribute [45]. Equalized Odds Difference (EOdiff) is a group fairness metric that assesses whether a predictive model achieves equal error rates across groups defined by a sensitive attribute, such as gender and race [46]. Specifically, it requires that the model's true positive rate (TPR) and false positive rate (FPR) be the same across all groups and is defined as the average of the absolute differences in TPR and FPR between groups [47].

4. Experimental Framework

In the current body of research on predictive modeling within the field of EDM, the evaluation of SSL algorithms is commonly conducted using datasets derived from existing labeled data. This is typically achieved by removing the class labels from a subset of an existing labeled dataset (D), thus creating a subset of unlabeled data (U) and a subset of labeled data (L), such that $U \cup D = L$. The proportion of labeled data (L) to the original data (D) is referred to as the labeled ratio (r).

4.1. Implementation of TVAE for Synthetic Data Generation

TVAE is a deep generative model that is particularly employed to create synthetic tabular data, typically including both continuous and categorical variables. Unlike standard variational autoencoders, TVAE has specific preprocessing procedures such as one-hot encoding for categorical variables and normalization for continuous variables to handle the mixed data types commonly encountered in tabular datasets. It learns a probabilistic latent representation of the data using an encoder, and samples are reconstructed using a decoder, with a loss function that trades off reconstruction quality against regularization of the latent space using the Kullback–Leibler divergence. Trained, TVAE can generate new synthetic data samples by sampling from the learned latent space, which is a valuable ingredient in data privacy, augmentation, and benchmarking in machine learning workflows.

TVAE is implemented as part of the open-source Synthetic Data Vault (SDV) library [47]. The SDV library's TVAE is designed to generate synthetic tabular data from learning the target dataset's latent representation. It follows an encoder–decoder architecture, where the encoder transforms input features into a lower-dimensional space and the decoder reconstructs data from the lower space. In this setup, latent embedding size is 128 and both compression and decompression networks have two hidden layers of 128 units. Regularization is taken care of by an L2 penalty of 0.00001 to prevent overfitting, and training proceeds with a batch size of 500 for 6 epochs. The model employs a loss factor of 2 for balancing reconstruction and regularization loss, and training is made to avail of GPU speedup if cuda = True. For maintaining data fidelity, the synthesizer imposes minimum–maximum value constraints on numerical features and rounding policies as needed. SingleTableMetadata is used to extract the metadata from the original dataset, with categorical and numerical features explicitly defined; e.g., the label column is treated as categorical. This structured metadata ensures the synthetic data respects the statistical characteristics and semantic meaning of the original table and facilitates TVAE in generating realistic and varied synthetic samples.

4.2. Description of the Proposed Algorithm

The preprocessing phase involves the normalization of continuous features and the transformation of categorical features using one-hot encoding. This step is critical for optimizing model performance and ensuring the stability of the training process.

Subsequently, the TVAE is trained on the labeled dataset, capturing the statistical distribution of the input data to later sample from it. The trained TVAE is employed to generate a synthetic dataset comprising N new labeled instances. To align with the SSL paradigm, the TVAE synthetic samples are treated as unlabeled by discarding their associated output labels. This transformation allows the synthetic data to augment the pool of unlabeled examples, thereby supporting the training of semi-supervised models without introducing label noise or bias. Next, the synthetic unlabeled dataset augments the original unlabeled dataset, and an SSL algorithm is trained for building a predictive model.

Algorithm 1 presents the pseudo-code of the proposed algorithm, while Figure 1 illustrates the overall workflow. Within each ratio, the evaluation proceeds as follows:

- For each labeled data proportion, the system enters an outer loop (Iterate Over Labeled Ratios).
- Within each ratio, K(=10)-fold cross-validation is performed. The dataset is partitioned into labeled and unlabeled subsets and preprocessed.
- The TVAE model is trained on the labeled subset to learn a generative representation. Synthetic samples X_S are generated from the trained TVAE. The unlabeled subset is augmented with the TVAE synthetic data.
- An SSL model is trained on the augmented dataset. The trained model is evaluated and performance metrics are recorded.
- The loop continues until all folds and all label ratios are processed.

Algorithm 1: TVAE SSL.

Input: LabeledData $\leftarrow$ small labeled dataset $(X_{\text{labeled}}, y_{\text{labeled}})$
UnlabeledData $\leftarrow$ large unlabeled dataset $X_{\text{unlabeled}}$
TVAEParameters $\leftarrow$ configuration parameters for the TVAE
$N_{\text{synthetic}} \leftarrow \text{len}(X_{\text{labeled}} + X_{\text{unlabeled}})/2$
SSLAlgorithm $\leftarrow$ SSL algorithm (e.g., Self-Training)
Classifier $\leftarrow$ base classifier (e.g., RandomForest)
Output: FinalModel $\leftarrow$ trained semi-supervised model

1 **Preprocess Data:** Normalize and encode X_{labeled} and $X_{\text{unlabeled}}$.;
2 **Train TVAE:** TVAE $\leftarrow$ TrainTVAE(X_{labeled}, TVAEParameters);
3 **Generate Synthetic Data:** SyntheticData $\leftarrow$ TVAE.GenerateSamples$\left(N_{\text{synthetic}}\right)$;
4 Discard pseudo-labels for *SyntheticData* to convert them into unlabeled.;
5 **Augment Training Set**;
6 AugmentedUnlabeledData $\leftarrow X_{\text{unlabeled}} \cup$ SyntheticData;
7 **Train Semi-Supervised Model:** FinalModel $\leftarrow$ TrainSSL(SSLAlgorithm(Classifier), LabeledData, AugmentedUnlabeledData);
8 **Evaluate Final Model:** Evaluate *FinalModel* on test set.;
9 Report performance metrics (accuracy, F1-score, etc.);

Our choice to synthesize artificial samples at half the dataset size was motivated by the need to balance augmentation and fidelity: synthesizing too many artificial samples relative to the real data risks overwhelming the SSL algorithms with lower-quality artificial data, while synthesizing too few has no appreciable effect on the training signal. The 50% ratio was therefore chosen as a compromise from early pilot experiments where larger ratios of synthetic data were not found to provide consistent improvements.

For training TVAE, we used a six-epoch setting. We did this for two reasons: (1) to prevent overfitting the generative model to small labeled subsets (especially in the lowest labeled ratio experiments), and (2) to keep computation feasible across a number of datasets, classifiers, and SSL methods. Interestingly, we observed empirically that training beyond

six epochs did not yield a significant difference in downstream SSL performance, which suggested that even with relatively limited training the generated samples provided useful additional structure for the learners.

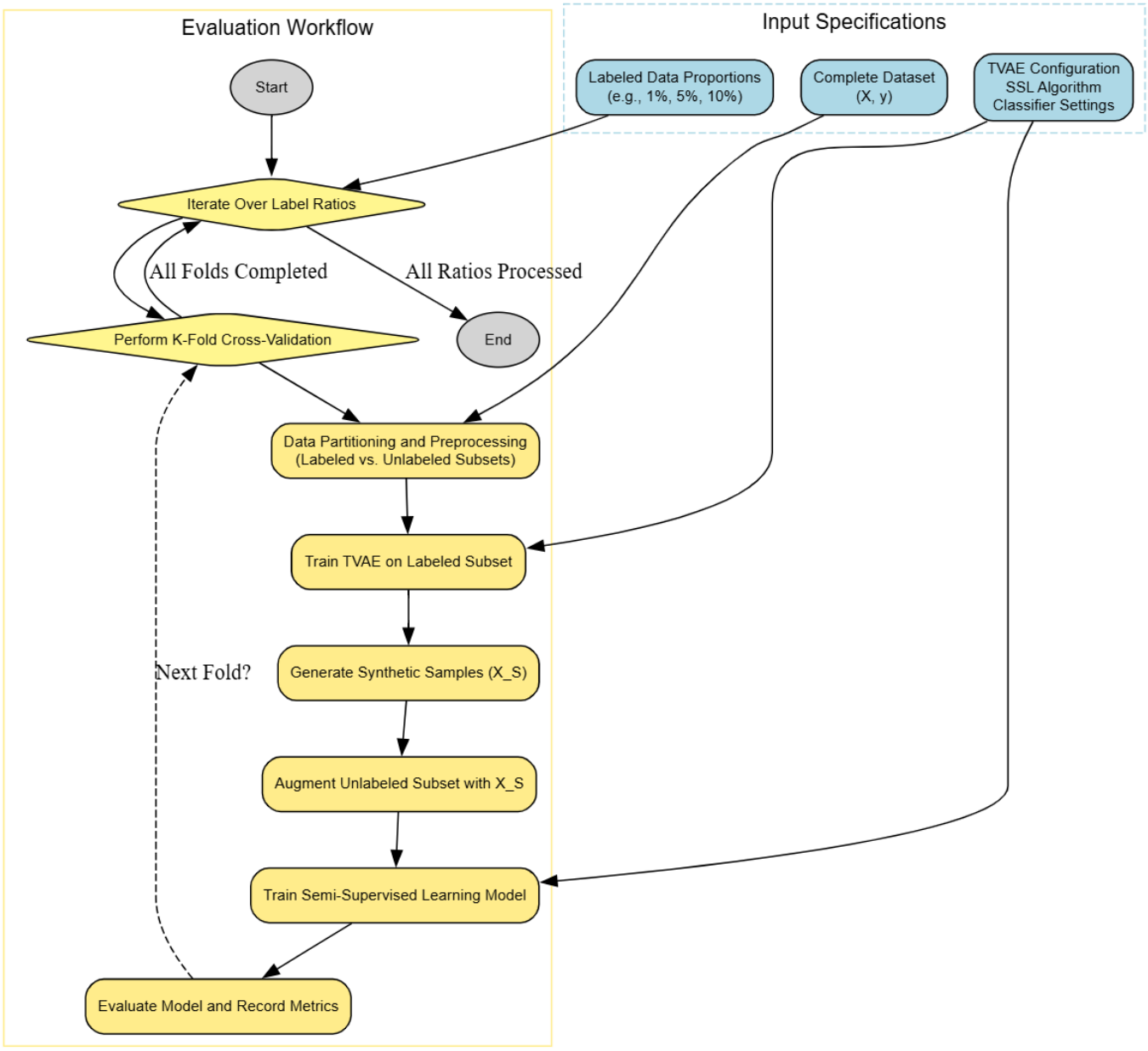

Figure 1. Workflow of the proposed SSL framework with TVAE augmentation.

5. Results

This section presents the performance and fairness results obtained from the experimental process, along with a corresponding discussion and analysis.

The experimental results across the four datasets demonstrate the competitive performance of SSL techniques compared to the fully supervised baseline (RF). Accuracy results listed in Table 6 show several significant trends across datasets and labeled ratios. In general, semi-supervised approaches provide significant improvements over fully supervised baseline (RF), particularly when significantly less than half of the data are labeled (e.g., ratios of 0.01 and 0.05). For the cert dataset, Tri-Training and Setred always perform better than others with best performance, especially at the lowest ratio where Tri-Training

achieves an accuracy of 0.8717. On the contrary, on eap2024, performance is relatively poor at a ratio of 0.01 but methods such as Rel-Rasco and CoBC show comparable gains with an increasing ratio, though CoBC does a slightly better job than others at a ratio of 0.3. The MBA dataset exhibits overall high and consistent accuracies for all semi-supervised methods, though CoBC and Self-Training showed the best consistent results, outperforming 0.83 even at very low ratios. On the harder student dataset, noticeable improvement in performance is observed when moving from 0.01 to higher ratios, and Rel-Rasco and CoBC are high performers at a ratio above 0.1, with accuracy measures over 0.85. The final Wins row also confirms these observations, where CoBC and Rel-Rasco dominate in the majority, followed by Self-Training and Rasco.

The F1-score figures in Table 7 provide a closer look at the performance of the models than accuracy, particularly in alleviating class imbalance. In the cert dataset, ensemble methods such as Tri-Training and Setred achieve top performance with the lowest ratio (0.01) F1-scores, whereas the performance is stabilized above 0.90 in nearly all the methods when more labeled data is available ($\geq$0.05). In the eap2024 dataset, performances are worse for very low ratios, but Rasco and Rel-Rasco stand out, consistently outperforming other approaches, especially for the ratios 0.1 and 0.2, where they reach above 0.76. The MBA dataset shows an interesting contrast between accuracy and F1-scores: whereas accuracy was high and stable, F1-scores were generally low (ranging from 0.31–0.41), reflecting extreme class imbalance, with Rasco and Rel-Rasco again achieving the highest values. On the student dataset, improvements are most marked as the ratio increases, with Rel-Rasco achieving the highest overall F1-scores above 0.83 at 0.2 and 0.3, with other ensemble methods like Tri-Training and CoBC also faring well. The Wins row corroborates this trend, with Rel-Rasco (6) and Rasco (5) well ahead by a large margin across datasets, indicating their stability in balancing precision and recall. These results highlight that while accuracy may reflect overall good performance, F1-scores reveal the merits of Rasco-based methods in being able to deal successfully with imbalanced or noisy scenarios.

The Demographic Parity Difference (DPD) values in Table 8 put the focus on vast discrepancies among datasets, methods, and labeled ratios. Lower DPD values are usually desirable because they represent more balanced models with less discrepancy in sensitive classes. In the cert dataset, all semi-supervised methods produce relatively low and similar values (mostly less than 0.10), while Co-Forest and CoBC tend to be closest to the lowest disparities, especially at smaller ratios. On the other hand, the eap2024 data shows higher variability, where methods such as Rel-Rasco yield much higher DPD (e.g., 0.2149 at the ratio 0.01), reflecting fairness problems, while Co-Forest and CoBC show smaller discrepancies. For MBA, values are all low across methods, where again Co-Forest and CoBC yield virtually zero values at small ratios, which is strong support for fairness robustness. The student set, however, exhibits much larger differences overall, with several methods (e.g., Setred, Co-Training) providing values above 0.20, whereas Self-Training sometimes provides very low DPD (even 0.0000 at the ratio 0.01), although not all the time. The last row win numbers confirm this pattern, where Self-Training is the overall winner most of the time (eight wins), followed by Co-Forest (four wins) and CoBC (three wins). Such results show that fairness performance is highly dataset-sensitive, with Self-Training being one of the best overall performers in minimizing demographic disparity, while some ensemble methods can achieve competitive performance for specific settings.

Table 6. Accuracy results (RF). Bold values indicate the highest value per row.

Dataset	Ratio	RF	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.8440 (0.0930)	0.8632 (0.0923)	0.8504 (0.0813)	0.8547 (0.0829)	0.8515 (0.0822)	0.8301 (0.0854)	**0.8717** (0.0971)	0.8440 (0.0915)
	0.05		0.9317 (0.0307)	0.9359 (0.0193)	0.9402 (0.0125)	**0.9434** (0.0133)	0.9370 (0.0137)	0.9263 (0.0223)	0.9391 (0.0213)	0.9295 (0.0230)
	0.1		0.9401 (0.0198)	**0.9455** (0.0116)	0.9359 (0.0149)	0.9391 (0.0150)	0.9391 (0.0182)	0.9444 (0.0150)	0.9380 (0.0150)	0.9348 (0.0212)
	0.2		0.9402 (0.0196)	0.9391 (0.0166)	0.9423 (0.0135)	0.9337 (0.0212)	0.9359 (0.0241)	**0.9455** (0.0197)	0.9434 (0.0175)	0.9423 (0.0215)
	0.3		**0.9423** (0.0215)	0.9348 (0.0222)	0.9391 (0.0201)	0.9359 (0.0207)	0.9337 (0.0224)	0.9359 (0.0220)	0.9412 (0.0162)	0.9402 (0.0208)
	1	**0.9466** (0.0246)								
eap2024	0.01		0.6161 (0.1477)	0.6720 (0.1015)	0.6001 (0.1410)	0.6769 (0.1129)	**0.7037** (0.0979)	0.6275 (0.1174)	0.6923 (0.0974)	0.5393 (0.0245)
	0.05		0.7348 (0.0648)	0.7482 (0.0549)	0.7171 (0.0601)	**0.7616** (0.0349)	0.7438 (0.0657)	0.7480 (0.0612)	0.7416 (0.0587)	0.7573 (0.0736)
	0.1		0.7369 (0.0790)	0.7414 (0.0542)	0.7055 (0.0841)	0.7683 (0.0420)	**0.7684** (0.0432)	0.7171 (0.0951)	0.7099 (0.0937)	0.7597 (0.0731)
	0.2		0.7571 (0.0469)	0.7437 (0.0502)	0.7277 (0.0532)	**0.7638** (0.0575)	0.7617 (0.0350)	0.7460 (0.0454)	0.7505 (0.0411)	0.7551 (0.0373)
	0.3		0.7908 (0.0336)	0.7774 (0.0283)	0.7415 (0.0591)	0.7819 (0.0393)	0.7708 (0.0654)	0.7977 (0.0430)	0.7887 (0.0377)	**0.7999** (0.0367)
	1	0.7774 (0.0529)								
MBA	0.01		**0.8374** (0.0039)	0.8240 (0.0158)	0.7386 (0.0499)	0.7798 (0.0386)	0.7891 (0.0339)	0.8339 (0.0071)	0.8311 (0.0079)	0.8356 (0.0049)
	0.05		**0.8371** (0.0033)	0.8258 (0.0144)	0.7399 (0.0178)	0.7806 (0.0154)	0.7925 (0.0190)	0.8335 (0.0085)	0.8268 (0.0131)	0.8356 (0.0062)
	0.1		0.8363 (0.0036)	0.8201 (0.0126)	0.7475 (0.0311)	0.7985 (0.0165)	0.7987 (0.0184)	0.8305 (0.0083)	0.8281 (0.0094)	**0.8368** (0.0040)
	0.2		0.8355 (0.0052)	0.8268 (0.0100)	0.7783 (0.0165)	0.8076 (0.0150)	0.8101 (0.0106)	0.8326 (0.0081)	0.8308 (0.0082)	**0.8381** (0.0050)
	0.3		0.8371 (0.0036)	0.8260 (0.0102)	0.7879 (0.0119)	0.8195 (0.0105)	0.8111 (0.0126)	0.8355 (0.0084)	0.8345 (0.0076)	**0.8376** (0.0037)
	1	0.8294 (0.0130)								
student	0.01		0.5608 (0.1395)	0.5992 (0.1158)	0.5820 (0.1302)	0.5356 (0.1473)	0.5818 (0.1075)	0.5785 (0.1225)	**0.6226** (0.0993)	0.4213 (0.1224)
	0.05		**0.8137** (0.1109)	0.7903 (0.1010)	0.6194 (0.0914)	0.6913 (0.1176)	0.7056 (0.1477)	0.7618 (0.0815)	0.7583 (0.0833)	0.7239 (0.0514)
	0.1		0.8224 (0.0760)	0.8514 (0.0764)	0.7179 (0.0625)	0.8052 (0.0435)	**0.8516** (0.0576)	0.8250 (0.0912)	0.8229 (0.0966)	0.8133 (0.0729)
	0.2		0.8634 (0.0686)	0.8545 (0.0660)	0.7354 (0.0630)	0.8169 (0.0778)	**0.8722** (0.0617)	0.8576 (0.0680)	0.8633 (0.0753)	0.8661 (0.0781)
	0.3		0.8576 (0.0881)	0.8546 (0.0755)	0.7704 (0.0618)	0.8050 (0.0602)	0.8576 (0.0654)	0.8575 (0.0913)	0.8489 (0.0786)	**0.8720** (0.0731)
	1	0.8402 (0.0535)								
	Wins	1	4	1	0	3	4	1	2	**5**

Table 7. F1-Score results (RF). Bold values indicate the highest value per row.

Dataset	Ratio	RF	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.6641 (0.2498)	0.7190 (0.2495)	0.7076 (0.2387)	0.7117 (0.2414)	0.7081 (0.2392)	0.6319 (0.2303)	**0.7309** (0.2578)	0.6647 (0.2477)
	0.05		0.9037 (0.0494)	0.9123 (0.0279)	0.9216 (0.0153)	**0.9256** (0.0164)	0.9158 (0.0182)	0.8964 (0.0363)	0.9171 (0.0316)	0.9015 (0.0393)
	0.1		0.9193 (0.0278)	**0.9278** (0.0139)	0.9168 (0.0175)	0.9203 (0.0174)	0.9201 (0.0219)	0.9255 (0.0183)	0.9177 (0.0186)	0.9135 (0.0282)
	0.2		0.9202 (0.0257)	0.9195 (0.0213)	0.9244 (0.0173)	0.9133 (0.0271)	0.9152 (0.0314)	**0.9279** (0.0255)	0.9247 (0.0236)	0.9238 (0.0277)
	0.3		**0.9230** (0.0288)	0.9132 (0.0295)	0.9203 (0.0254)	0.9149 (0.0273)	0.9115 (0.0302)	0.9137 (0.0301)	0.9220 (0.0210)	0.9214 (0.0259)
	1	**0.9290** (0.0319)								
eap2024	0.01		0.5118 (0.2312)	0.6196 (0.1757)	0.5577 (0.1777)	0.6357 (0.1761)	**0.6639** (0.1714)	0.5400 (0.1993)	0.6508 (0.1684)	0.3694 (0.0512)
	0.05		0.7304 (0.0700)	0.7432 (0.0580)	0.7123 (0.0617)	**0.7597** (0.0352)	0.7391 (0.0672)	0.7428 (0.0638)	0.7321 (0.0696)	0.7492 (0.0791)
	0.1		0.7318 (0.0817)	0.7374 (0.0539)	0.6966 (0.0923)	**0.7676** (0.0418)	0.7673 (0.0433)	0.7038 (0.1090)	0.7012 (0.0980)	0.7542 (0.0758)
	0.2		0.7540 (0.0475)	0.7410 (0.0521)	0.7227 (0.0549)	**0.7614** (0.0577)	0.7601 (0.0351)	0.7439 (0.0461)	0.7482 (0.0412)	0.7513 (0.0393)
	0.3		0.7891 (0.0333)	0.7763 (0.0287)	0.7381 (0.0625)	0.7807 (0.0402)	0.7701 (0.0660)	0.7951 (0.0438)	0.7875 (0.0378)	**0.7986** (0.0366)
	1	0.7764 (0.0526)								
MBA	0.01		0.3160 (0.0240)	0.3575 (0.0392)	**0.4016** (0.0289)	0.3861 (0.0378)	0.3952 (0.0358)	0.3135 (0.0208)	0.3457 (0.0348)	0.3166 (0.0273)
	0.05		0.3186 (0.0303)	0.3708 (0.0213)	0.3999 (0.0241)	0.4050 (0.0232)	**0.4121** (0.0167)	0.3317 (0.0399)	0.3440 (0.0249)	0.3213 (0.0317)
	0.1		0.3194 (0.0154)	0.3739 (0.0349)	0.3888 (0.0216)	0.4129 (0.0234)	**0.4143** (0.0178)	0.3328 (0.0276)	0.3511 (0.0191)	0.3251 (0.0259)
	0.2		0.3129 (0.0122)	0.3913 (0.0257)	0.3902 (0.0234)	**0.4106** (0.0159)	0.4078 (0.0175)	0.3390 (0.0159)	0.3569 (0.0272)	0.3344 (0.0289)
	0.3		0.3253 (0.0222)	0.3892 (0.0280)	0.3821 (0.0170)	0.4037 (0.0338)	**0.4146** (0.0157)	0.3536 (0.0221)	0.3636 (0.0169)	0.3321 (0.0204)
	1	0.3949 (0.0207)								
student	0.01		0.3543 (0.0623)	0.4820 (0.0980)	0.5146 (0.1327)	0.4703 (0.1370)	0.5162 (0.1293)	0.3917 (0.0737)	**0.5295** (0.1184)	0.2989 (0.0542)
	0.05		**0.7500** (0.2006)	0.7409 (0.1555)	0.5767 (0.1086)	0.6540 (0.1450)	0.6641 (0.1742)	0.7131 (0.1322)	0.6898 (0.1599)	0.6228 (0.1358)
	0.1		0.7885 (0.0984)	0.8287 (0.0930)	0.6791 (0.0720)	0.7788 (0.0586)	**0.8311** (0.0695)	0.7851 (0.1228)	0.7951 (0.1173)	0.7766 (0.0961)
	0.2		0.8411 (0.0830)	0.8315 (0.0811)	0.6978 (0.0748)	0.7824 (0.1127)	**0.8560** (0.0725)	0.8336 (0.0857)	0.8393 (0.0928)	0.8411 (0.0963)
	0.3		0.8383 (0.1009)	0.8331 (0.0919)	0.7289 (0.0869)	0.7718 (0.0802)	0.8386 (0.0764)	0.8379 (0.1043)	0.8287 (0.0917)	**0.8535** (0.0850)
	1	0.8180 (0.0632)								
	Wins	1	2	1	1	5	**6**	1	2	2

Table 8. Demographic Parity Difference results (RF). Bold values indicate the lowest value per row.

Dataset	Ratio	RF	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		**0.0186** (0.0329)	0.0420 (0.0663)	0.0452 (0.0641)	0.0481 (0.0760)	0.0457 (0.0739)	0.0314 (0.0529)	0.0425 (0.0831)	0.0298 (0.0383)
	0.05		0.0921 (0.0815)	**0.0876** (0.0905)	0.0994 (0.0979)	0.1029 (0.0953)	0.0992 (0.0832)	0.0902 (0.0813)	0.0932 (0.0913)	0.0993 (0.0809)
	0.1		0.0898 (0.0914)	0.0922 (0.0948)	0.0855 (0.0787)	0.0947 (0.0855)	0.0910 (0.0887)	**0.0715** (0.0769)	0.0817 (0.0842)	0.0868 (0.0911)
	0.2		0.0930 (0.0849)	0.0990 (0.0917)	0.0961 (0.0941)	0.0940 (0.0992)	0.0916 (0.0952)	**0.0847** (0.1020)	0.0979 (0.0923)	0.0932 (0.0969)
	0.3		0.1024 (0.0945)	0.0941 (0.0948)	0.1007 (0.1009)	0.0938 (0.0947)	0.1024 (0.0963)	**0.0883** (0.0877)	0.0956 (0.0932)	0.0927 (0.0972)
	1	0.0960 (0.0955)								
eap2024	0.01		0.0664 (0.0913)	0.1252 (0.1141)	0.1212 (0.1293)	0.0655 (0.0943)	0.2149 (0.1754)	0.0556 (0.0741)	0.1166 (0.1240)	**0.0310** (0.0686)
	0.05		**0.0938** (0.0501)	0.1320 (0.0727)	0.1639 (0.1014)	0.1174 (0.0737)	0.1771 (0.0959)	0.1525 (0.1280)	0.1332 (0.0934)	0.1086 (0.0749)
	0.1		0.1739 (0.1097)	0.1137 (0.0745)	0.1405 (0.0658)	0.1251 (0.1061)	0.1558 (0.1658)	**0.0983** (0.0696)	0.1343 (0.0957)	0.1774 (0.1063)
	0.2		**0.1000** (0.0962)	0.1140 (0.1007)	0.1531 (0.1177)	0.1513 (0.0512)	0.1267 (0.1022)	0.1256 (0.0938)	0.1291 (0.0686)	0.1134 (0.1029)
	0.3		0.1295 (0.1057)	**0.1011** (0.1013)	0.1871 (0.1233)	0.1426 (0.0791)	0.1119 (0.0792)	0.1238 (0.0837)	0.1112 (0.0800)	0.1382 (0.1153)
	1	0.1795 (0.0831)								
MBA	0.01		0.0039 (0.0066)	0.0141 (0.0133)	0.0590 (0.0521)	0.0192 (0.0133)	0.0190 (0.0158)	0.0023 (0.0039)	0.0088 (0.0072)	**0.0019** (0.0037)
	0.05		**0.0029** (0.0050)	0.0305 (0.0185)	0.0653 (0.0408)	0.0285 (0.0200)	0.0295 (0.0142)	0.0098 (0.0143)	0.0222 (0.0221)	0.0051 (0.0075)
	0.1		**0.0060** (0.0068)	0.0408 (0.0216)	0.0813 (0.0417)	0.0627 (0.0565)	0.0324 (0.0141)	0.0134 (0.0170)	0.0320 (0.0206)	0.0116 (0.0129)
	0.2		**0.0060** (0.0080)	0.0447 (0.0161)	0.0677 (0.0224)	0.0270 (0.0115)	0.0242 (0.0287)	0.0198 (0.0123)	0.0373 (0.0138)	0.0170 (0.0149)
	0.3		**0.0114** (0.0100)	0.0429 (0.0128)	0.0587 (0.0169)	0.0279 (0.0240)	0.0166 (0.0136)	0.0254 (0.0140)	0.0320 (0.0133)	0.0170 (0.0176)
	1	0.0463 (0.0111)								
student	0.01		**0.0000** (0.0000)	0.1134 (0.1568)	0.1201 (0.1331)	0.0728 (0.0868)	0.0574 (0.0779)	0.0282 (0.0396)	0.1110 (0.1071)	0.0091 (0.0198)
	0.05		0.1036 (0.1108)	0.1484 (0.1150)	0.2750 (0.2042)	0.2084 (0.1645)	**0.0985** (0.0822)	0.1991 (0.1532)	0.1023 (0.0932)	0.1001 (0.1127)
	0.1		0.1519 (0.0671)	0.1646 (0.0875)	0.1515 (0.1499)	0.1434 (0.1224)	0.1665 (0.0956)	0.1733 (0.1060)	0.1779 (0.0832)	**0.1415** (0.1216)
	0.2		0.1310 (0.0911)	0.1720 (0.0689)	**0.1200** (0.1040)	0.1849 (0.1422)	0.1520 (0.0841)	0.1651 (0.1008)	0.1720 (0.0780)	0.1693 (0.0889)
	0.3		0.1762 (0.1034)	0.2016 (0.1010)	**0.0952** (0.0993)	0.1443 (0.0793)	0.1726 (0.0633)	0.1617 (0.0935)	0.1735 (0.0792)	0.1412 (0.0646)
	1	0.1454 (0.0656)								
	Wins	0	**8**	2	2	0	1	4	0	3

The Equalized Odds Difference (EOD) values in Table 9 indicate that fairness disparities are extremely dataset-specific and typically larger than those of Demographic Parity. For the cert dataset, a majority of algorithms exhibit relatively small disparities (frequently

smaller than 0.12), with Rasco, Tri-Training, and Co-Training frequently possessing the smallest ones, especially at low labeled ratios. On the other hand, the eap2024 dataset shows the largest disparities uniformly across methods with values often above 0.20–0.30, indicating that it is significantly more difficult to enforce fairness in the Equalized Odds sense in this setting. The MBA dataset produces the best outcome, where Self-Training, Co-Forest, and CoBC show very small disparities (almost zero for low ratios), which is indicative of their capability to enforce fairness. The student set, however, demonstrates the strongest challenges, where high imbalances in almost all methods (typically above 0.25–0.35) reflect difficulty in balancing false positive and false negative rates among groups. The win counts reflect this pattern: Self-Training takes the top with eight wins, with Co-Training and CoBC each tied with three wins. This means that although there is no method guaranteeing low disparities on every dataset, Self-Training always produces more balanced outcomes under Equalized Odds, though its efficiency might vary significantly depending on the data complexity.

As shown in Figure 2, certain semi-supervised methods outperform others when Random Forest is used as the base classifier, indicating differences in their adaptability to the datasets.

Similarly, the experimental results across the four datasets demonstrate the competitive performance of SSL techniques compared to the fully supervised baseline (HGB).

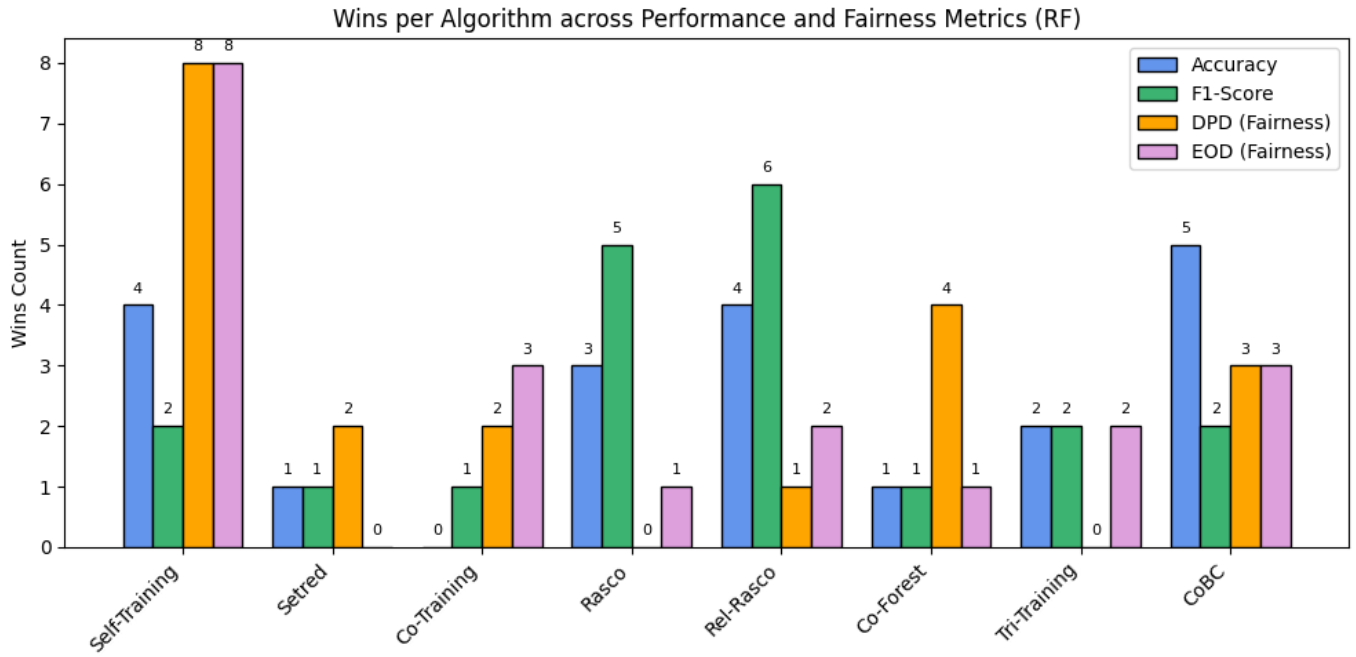

Figure 2. Summary of performance across SSL algorithms using RF as the base classifier.

The accuracy results in Table 10 reveal several interesting trends for the performance of different SSL methods on the four datasets and varying labeled ratios. For the cert dataset, the majority of the methods display a clear improvement with the increasing labeled ratio, with Rel-Rasco and CoBC always performing best at small ratios, and the performance of methods converges with more available labeled instances. In comparison, the eap2024 dataset is more challenging to work with, with all techniques achieving only baseline-level accuracy at 0.01 and 0.05 ratios but significant improvements at >0.1, where ensemble-based algorithms such as Rasco and Co-Forest become dominant. The MBA dataset is relatively consistent and shows high performance by all techniques even at very low labeled ratios, with CoBC being slightly better than the rest in most cases. Finally, in the student dataset,

accuracies start quite low for low ratios but rapidly catch up, and Rel-Rasco and CoBC clearly outperform others with higher labeled ratios.

Table 9. Equalized Odds Difference results (RF). Bold values indicate the lowest value per row.

Dataset	Ratio	RF	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.0373 (0.0562)	0.0438 (0.0687)	0.0268 (0.0389)	**0.0193** (0.0269)	0.0237 (0.0335)	0.0611 (0.1153)	0.0239 (0.0298)	0.0529 (0.0659)
	0.05		0.1177 (0.0969)	0.0840 (0.0687)	**0.0815** (0.0790)	0.0921 (0.0824)	0.1230 (0.1040)	0.1243 (0.0948)	0.0849 (0.0531)	0.1219 (0.0648)
	0.1		0.0967 (0.0749)	0.0998 (0.0500)	0.0702 (0.0636)	0.1066 (0.0559)	0.0869 (0.0582)	0.0866 (0.0513)	**0.0660** (0.0367)	0.0834 (0.0525)
	0.2		0.1148 (0.0668)	0.1234 (0.0780)	0.1106 (0.0345)	0.0992 (0.0632)	0.1056 (0.0842)	0.0944 (0.0465)	0.0954 (0.0470)	**0.0821** (0.0428)
	0.3		0.1147 (0.0847)	0.1191 (0.1018)	**0.1056** (0.0520)	0.1184 (0.0852)	0.1170 (0.0771)	0.1209 (0.0749)	0.1278 (0.0674)	0.1196 (0.0587)
	1	0.1108 (0.0849)								
eap2024	0.01		0.0865 (0.1375)	0.2124 (0.2068)	0.2371 (0.1761)	0.2307 (0.1602)	0.2631 (0.2409)	0.1275 (0.1576)	0.2839 (0.2222)	**0.0667** (0.1440)
	0.05		**0.2031** (0.0961)	0.2360 (0.1256)	0.3045 (0.1301)	0.2845 (0.1324)	0.3338 (0.1384)	0.2734 (0.1595)	0.2247 (0.1328)	0.2359 (0.1521)
	0.1		**0.2364** (0.1272)	0.2468 (0.1275)	0.3184 (0.1097)	0.2913 (0.1544)	0.3001 (0.1512)	0.2490 (0.1048)	0.2696 (0.1315)	0.2720 (0.0980)
	0.2		**0.2059** (0.0906)	0.2592 (0.0980)	0.3185 (0.1295)	0.2561 (0.0843)	0.2622 (0.0819)	0.2257 (0.0984)	0.2172 (0.0842)	0.2236 (0.0752)
	0.3		0.2669 (0.1059)	0.2481 (0.0999)	0.2881 (0.1980)	**0.2125** (0.1537)	0.2560 (0.0840)	0.2343 (0.1425)	0.2263 (0.1179)	0.1997 (0.1004)
	1	0.2625 (0.1135)								
MBA	0.01		0.0138 (0.0300)	0.0380 (0.0234)	0.0874 (0.0394)	0.0466 (0.0469)	0.0916 (0.0587)	**0.0128** (0.0221)	0.0342 (0.0306)	0.0149 (0.0270)
	0.05		**0.0170** (0.0353)	0.0761 (0.0450)	0.1071 (0.0347)	0.0632 (0.0385)	0.0961 (0.0466)	0.0339 (0.0574)	0.0521 (0.0474)	0.0185 (0.0351)
	0.1		**0.0177** (0.0161)	0.0838 (0.0604)	0.1184 (0.0545)	0.0969 (0.1036)	0.0891 (0.0530)	0.0276 (0.0301)	0.0707 (0.0465)	0.0264 (0.0266)
	0.2		**0.0136** (0.0115)	0.0864 (0.0615)	0.0706 (0.0274)	0.0630 (0.0482)	0.1045 (0.0669)	0.0412 (0.0178)	0.0841 (0.0443)	0.0455 (0.0490)
	0.3		**0.0260** (0.0197)	0.0774 (0.0420)	0.0746 (0.0256)	0.0685 (0.0577)	0.0759 (0.0461)	0.0489 (0.0295)	0.0645 (0.0347)	0.0410 (0.0373)
	1	0.0736 (0.0384)								
student	0.01		**0.0000** (0.0000)	0.1968 (0.2223)	0.2604 (0.2068)	0.1542 (0.1400)	0.1789 (0.1907)	0.0569 (0.1046)	0.2273 (0.1583)	0.0136 (0.0296)
	0.05		0.2291 (0.2060)	0.2686 (0.2007)	0.3950 (0.2250)	0.3179 (0.2190)	0.2718 (0.1849)	0.3485 (0.2919)	**0.2214** (0.1222)	0.2260 (0.1219)
	0.1		0.2769 (0.1652)	0.2710 (0.1704)	0.3551 (0.1924)	0.2470 (0.1537)	**0.2324** (0.0908)	0.3035 (0.1954)	0.2643 (0.1717)	0.2550 (0.2073)
	0.2		0.2461 (0.2102)	0.2851 (0.1335)	0.3598 (0.1797)	0.2344 (0.1539)	**0.2329** (0.1358)	0.3188 (0.2468)	0.3202 (0.1809)	0.3329 (0.2462)
	0.3		0.3024 (0.1970)	0.3442 (0.2134)	**0.2390** (0.1936)	0.2946 (0.1433)	0.2764 (0.1980)	0.2696 (0.1873)	0.3029 (0.1963)	0.2471 (0.1974)
	1	0.2520 (0.1564)								
	Wins	0	**8**	0	3	1	2	1	2	3

Table 10. Accuracy results (HGB). Bold values indicate the highest value per row.

Dataset	Ratio	HGB	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.7575 (0.0042)	0.7575 (0.0042)	0.7864 (0.0657)	0.8493 (0.0803)	**0.8568** (0.0862)	0.7070 (0.1615)	0.7575 (0.0042)	0.7723 (0.0204)
	0.05		0.8214 (0.0757)	0.7777 (0.0667)	0.9231 (0.0157)	0.9273 (0.0195)	**0.9359** (0.0181)	0.8269 (0.0661)	0.8375 (0.0697)	0.8077 (0.0767)
	0.1		0.9242 (0.0258)	0.9231 (0.0284)	0.9327 (0.0267)	**0.9391** (0.0195)	0.9380 (0.0219)	0.8976 (0.0520)	0.9306 (0.0276)	0.9327 (0.0239)
	0.2		0.9241 (0.0330)	0.9230 (0.0289)	**0.9433** (0.0153)	0.9295 (0.0220)	0.9316 (0.0183)	0.9188 (0.0221)	0.9295 (0.0258)	0.9338 (0.0288)
	0.3		0.9252 (0.0169)	0.9337 (0.0143)	0.9305 (0.0169)	0.9316 (0.0197)	0.9274 (0.0193)	**0.9358** (0.0273)	0.9295 (0.0198)	0.9316 (0.0210)
	1	0.9423 (0.0242)								
eap2024	0.01		**0.5168** (0.0243)	**0.5168** (0.0243)	**0.5168** (0.0243)	**0.5168** (0.0243)	**0.5168** (0.0243)	**0.5168** (0.0243)	0.5102 (0.0281)	0.5126 (0.0419)
	0.05		0.5168 (0.0243)	0.5168 (0.0243)	**0.5821** (0.0875)	0.5576 (0.0705)	0.5282 (0.0451)	0.5102 (0.0281)	0.5077 (0.0289)	0.5576 (0.0521)
	0.1		0.7080 (0.0846)	0.7307 (0.0838)	0.7394 (0.0308)	**0.7775** (0.0405)	0.7660 (0.0504)	0.6821 (0.0853)	0.7706 (0.0518)	0.7551 (0.0494)
	0.2		0.7866 (0.0438)	**0.7909** (0.0395)	0.7686 (0.0574)	0.7842 (0.0612)	0.7822 (0.0456)	0.7552 (0.0448)	0.7888 (0.0298)	0.7820 (0.0396)
	0.3		0.7752 (0.0402)	0.7798 (0.0515)	0.7687 (0.0528)	**0.8156** (0.0427)	0.7978 (0.0365)	0.8021 (0.0495)	0.7932 (0.0520)	0.8066 (0.0293)
	1	0.7842 (0.0437)								
MBA	0.01		0.8092 (0.0261)	0.8103 (0.0257)	0.7457 (0.0374)	0.7782 (0.0352)	0.7581 (0.0489)	0.8153 (0.0222)	0.8095 (0.0294)	**0.8211** (0.0190)
	0.05		0.8024 (0.0146)	0.7998 (0.0188)	0.7540 (0.0146)	0.7782 (0.0325)	0.7938 (0.0198)	0.7985 (0.0238)	0.8095 (0.0136)	**0.8295** (0.0136)
	0.1		0.8126 (0.0178)	0.8040 (0.0213)	0.7653 (0.0261)	0.7954 (0.0176)	0.7937 (0.0190)	0.8085 (0.0130)	0.8090 (0.0203)	**0.8242** (0.0152)
	0.2		0.8161 (0.0106)	0.8029 (0.0093)	0.7853 (0.0132)	0.8101 (0.0166)	0.8076 (0.0053)	0.8164 (0.0136)	0.8103 (0.0094)	**0.8260** (0.0095)
	0.3		0.8197 (0.0125)	0.8079 (0.0083)	0.7964 (0.0153)	0.8155 (0.0139)	0.8114 (0.0119)	0.8166 (0.0052)	0.8132 (0.0062)	**0.8263** (0.0069)
	1	0.8213 (0.0119)								
student	0.01		**0.5608** (0.1395)	**0.5608** (0.1395)	**0.5608** (0.1395)	**0.5608** (0.1395)	**0.5608** (0.1395)	**0.5608** (0.1395)	0.5000 (0.1536)	0.5491 (0.1391)
	0.05		**0.6160** (0.0929)	**0.6160** (0.0929)	**0.6160** (0.0929)	**0.6160** (0.0929)	0.6131 (0.0920)	0.5608 (0.1395)	**0.6160** (0.0929)	0.6155 (0.1053)
	0.1		0.6454 (0.0098)	0.6454 (0.0098)	0.6800 (0.0485)	0.7411 (0.0749)	**0.7441** (0.0682)	0.5961 (0.1250)	0.6454 (0.0098)	0.6683 (0.0513)
	0.2		0.8142 (0.0735)	0.7824 (0.1282)	0.7850 (0.0586)	0.8402 (0.0616)	**0.8718** (0.0930)	0.8108 (0.0699)	0.8022 (0.1099)	0.8112 (0.0913)
	0.3		0.8573 (0.0765)	0.8545 (0.0783)	0.8055 (0.0766)	0.8519 (0.0602)	**0.8693** (0.0631)	0.8429 (0.0768)	0.8546 (0.0730)	0.8632 (0.0758)
	1	0.8402 (0.0536)								
	Wins	0	3	1	2	3	**5**	1	0	**5**

The F1-score results in Table 11 provide further insight into the performance of the SSL methods under different levels of labeled data availability. On the cert dataset, Rel-Rasco and Rasco achieve the highest scores, particularly at low labeled ratios, reflecting their effectiveness in situations where there is limited supervision. The more challenging eap2024 dataset exhibits extremely poor performance at the lowest ratios across all methods, with just those from 0.1 onwards registering discernible gains. Rasco and Rel-Rasco shine here again, with competitive F1-scores and superior performance at higher ratios compared to the rest. The MBA dataset, on the other hand, shows a different trend: F1-scores are generally low throughout all ratios, with only minor increases when increasing amounts of labeled data are added, indicating that this dataset is intrinsically harder for semi-supervised methods. The student dataset, lastly, exhibits a significant increase in F1-score for larger labeled ratios, where Rel-Rasco and CoBC convincingly lead, recording the highest values and proving themselves well adaptable to more informative labeled subsets. Overall, Rel-Rasco is the most consistent top performer across datasets, achieving the most wins, followed by Rasco and then CoBC, confirming the usefulness of ensemble-based solutions to semi-supervised learning.

The Demographic Parity Difference (DPD) values in Table 12 have high variation across datasets, methods, and labeled ratios. At extremely low ratios (0.01), all methods achieve perfect demographic parity (DPD = 0), particularly in the eap2024 and student datasets, whereas methods such as CoBC at times display minor deviations. As the labeled ratio increases, disparities generally grow, with student and eap2024 recording the highest DPD values (with most exceeding 0.15 at ratios ($\geq$0.2)), indicating greater bias amplification in the low-supervision scenario. By contrast, the MBA dataset consistently shows relatively low DPD across methods (with most below 0.07), suggesting that fairness is less susceptible in this domain. Methodologically, Self-Training and Rel-Rasco tend to generate the lowest DPD scores, reflected in their highest number of wins, while ensemble-based approaches like Co-Training and Rasco sometimes bring in greater differences.

Equalized Odds Difference (EOD) measures in Table 13 reveal larger fairness problems than Demographic Parity Difference. At extremely low labeling rates (0.01 and 0.05), most techniques have almost perfect fairness (EOD = 0) for the eap2024 and student data, though some techniques (e.g., CoBC) already exhibit non-trivial differences. As the ratio increases, however, differences suddenly increase, especially in eap2024 and student, where many methods have over 0.25, indicating rather large differences across groups in false positive and false negative rates. The MBA dataset, in contrast, consistently reports lower EOD values (largely below 0.15), whereas cert has moderate differences between methods. From the viewpoint of method performance, Self-Training achieves the optimum count of wins, tending to register smaller differences across datasets, while ensemble methods such as Co-Training, Rasco, and Tri-Training tend to register larger values of EOD. Overall, these results suggest that ensuring fairness with respect to equalized odds is significantly more challenging compared to demographic parity, with both dataset type and choice of method playing crucial roles in the level of bias observed.

As shown in Figure 3, certain semi-supervised methods outperform others when HGB is used as the base classifier, indicating differences in their adaptability to the datasets.

Similarly, the experimental results across the four datasets demonstrate the competitive performance of SSL techniques compared to the fully supervised baseline (XGB).

Table 11. F1-Score results (HGB). Bold values indicate the highest value per row.

Dataset	Ratio	HGB	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.4310 (0.0014)	0.4310 (0.0014)	0.5768 (0.1706)	0.7024 (0.2346)	**0.7128** (0.2432)	0.4079 (0.0737)	0.4310 (0.0014)	0.4906 (0.0800)
	0.05		0.6540 (0.1894)	0.5842 (0.1519)	0.9002 (0.0186)	0.9023 (0.0279)	**0.9154** (0.0220)	0.6667 (0.1873)	0.7177 (0.1621)	0.6475 (0.1718)
	0.1		0.9000 (0.0341)	0.8971 (0.0386)	0.9103 (0.0357)	**0.9188** (0.0250)	0.9179 (0.0276)	0.8442 (0.0978)	0.9064 (0.0386)	0.9127 (0.0307)
	0.2		0.8984 (0.0457)	0.8974 (0.0393)	**0.9246** (0.0202)	0.9077 (0.0269)	0.9101 (0.0235)	0.8908 (0.0283)	0.9063 (0.0357)	0.9122 (0.0379)
	0.3		0.8984 (0.0242)	0.9094 (0.0217)	0.9073 (0.0220)	0.9089 (0.0267)	0.9028 (0.0260)	**0.9122** (0.0378)	0.9050 (0.0285)	0.9086 (0.0285)
	1	0.9222 (0.0342)								
eap2024	0.01		0.3406 (0.0108)	0.3406 (0.0108)	0.3406 (0.0108)	0.3406 (0.0108)	0.3406 (0.0108)	0.3406 (0.0108)	0.3376 (0.0125)	**0.3638** (0.0624)
	0.05		0.3406 (0.0108)	0.3406 (0.0108)	**0.4624** (0.1549)	0.4178 (0.1357)	0.3643 (0.0769)	0.3376 (0.0125)	0.3365 (0.0128)	0.4266 (0.1034)
	0.1		0.6833 (0.1338)	0.7105 (0.1359)	0.7338 (0.0347)	**0.7761** (0.0410)	0.7651 (0.0510)	0.6582 (0.1315)	0.7680 (0.0543)	0.7494 (0.0519)
	0.2		0.7856 (0.0440)	**0.7899** (0.0390)	0.7647 (0.0604)	0.7814 (0.0630)	0.7800 (0.0467)	0.7535 (0.0450)	0.7878 (0.0300)	0.7808 (0.0403)
	0.3		0.7742 (0.0405)	0.7791 (0.0520)	0.7667 (0.0543)	**0.8141** (0.0441)	0.7973 (0.0368)	0.8007 (0.0499)	0.7924 (0.0525)	0.8056 (0.0296)
	1	0.7831 (0.0441)								
MBA	0.01		0.3810 (0.0546)	0.3818 (0.0552)	0.4019 (0.0313)	**0.4073** (0.0295)	0.3951 (0.0366)	0.3590 (0.0359)	0.3838 (0.0396)	0.3726 (0.0475)
	0.05		0.4041 (0.0386)	0.4065 (0.0415)	0.4055 (0.0139)	0.3942 (0.0316)	**0.4164** (0.0312)	0.3907 (0.0319)	0.4081 (0.0353)	0.3773 (0.0361)
	0.1		0.3938 (0.0193)	0.4026 (0.0268)	0.4059 (0.0255)	0.4034 (0.0184)	**0.4197** (0.0264)	0.3887 (0.0220)	0.3967 (0.0224)	0.3816 (0.0171)
	0.2		0.3912 (0.0216)	0.4001 (0.0076)	0.4003 (0.0140)	**0.4189** (0.0188)	0.4064 (0.0300)	0.3901 (0.0105)	0.3957 (0.0121)	0.3848 (0.0133)
	0.3		0.3950 (0.0107)	0.4063 (0.0137)	0.3938 (0.0201)	0.3894 (0.0306)	**0.4112** (0.0134)	0.3994 (0.0203)	0.3981 (0.0142)	0.3827 (0.0141)
	1	0.3966 (0.0151)								
student	0.01		0.3543 (0.0623)	0.3543 (0.0623)	0.3543 (0.0623)	0.3543 (0.0623)	0.3543 (0.0623)	0.3543 (0.0623)	0.3270 (0.0689)	**0.3645** (0.0684)
	0.05		0.3790 (0.0417)	0.3790 (0.0417)	0.3790 (0.0417)	0.3790 (0.0417)	0.3780 (0.0413)	0.3543 (0.0623)	0.3790 (0.0417)	**0.4444** (0.0827)
	0.1		0.3922 (0.0036)	0.3922 (0.0036)	0.5100 (0.1360)	0.6558 (0.1555)	**0.6793** (0.1387)	0.3832 (0.0790)	0.3922 (0.0036)	0.4971 (0.1094)
	0.2		0.7917 (0.0827)	0.7690 (0.1298)	0.7536 (0.0725)	0.8162 (0.0744)	**0.8525** (0.1077)	0.7829 (0.0911)	0.7806 (0.1164)	0.7935 (0.0937)
	0.3		0.8356 (0.0912)	0.8350 (0.0937)	0.7792 (0.0838)	0.8306 (0.0716)	**0.8525** (0.0733)	0.8217 (0.0885)	0.8348 (0.0888)	0.8440 (0.0878)
	1	0.8177 (0.0658)								
	Wins	0	0	1	2	5	**8**	1	0	3

Table 12. Demographic Parity Difference results (HGB). Bold values indicate the lowest value per row.

Dataset	Ratio	HGB	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	0.0421 (0.0627)	0.0555 (0.0740)	0.0532 (0.0722)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0113 (0.0155)
	0.05		**0.0473** (0.0394)	0.0638 (0.0505)	0.0981 (0.0712)	0.1072 (0.0860)	0.1067 (0.1180)	0.0549 (0.0680)	0.0806 (0.0743)	0.0628 (0.0693)
	0.1		0.0899 (0.0943)	0.0943 (0.0911)	0.0942 (0.0927)	0.1099 (0.0937)	0.1033 (0.0891)	**0.0836** (0.0811)	0.0996 (0.0972)	0.0859 (0.0789)
	0.2		0.0966 (0.0899)	0.0975 (0.0871)	0.0994 (0.0859)	0.1011 (0.0942)	**0.0860** (0.0867)	0.0893 (0.0766)	0.0955 (0.0955)	0.0990 (0.0955)
	0.3		0.0891 (0.0864)	0.0904 (0.0861)	0.1043 (0.1011)	0.1023 (0.0995)	**0.0883** (0.0814)	0.0924 (0.0865)	0.0918 (0.0934)	0.0992 (0.0929)
	1	0.0969 (0.0690)								
eap2024	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0272 (0.0432)
	0.05		**0.0000** (0.0000)	**0.0000** (0.0000)	0.0745 (0.0905)	0.0398 (0.0951)	0.0320 (0.1013)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0740 (0.0955)
	0.1		0.1121 (0.0809)	0.1476 (0.1220)	0.1589 (0.1316)	0.1564 (0.1052)	0.1450 (0.1537)	0.1128 (0.0758)	**0.1002** (0.0905)	0.1340 (0.0716)
	0.2		0.1244 (0.0711)	0.1225 (0.1183)	0.1395 (0.1059)	0.1233 (0.1004)	**0.1189** (0.1029)	0.1633 (0.1023)	0.1256 (0.1002)	0.1358 (0.1165)
	0.3		0.1083 (0.0930)	0.1184 (0.1048)	0.1740 (0.1032)	0.1182 (0.1121)	**0.1029** (0.0951)	0.1465 (0.0866)	0.1272 (0.0742)	0.1310 (0.1056)
	1	0.1845 (0.0953)								
MBA	0.01		0.0159 (0.0093)	0.0247 (0.0304)	0.0521 (0.0318)	0.0450 (0.0171)	0.0460 (0.0545)	**0.0100** (0.0063)	0.0184 (0.0142)	0.0116 (0.0128)
	0.05		0.0602 (0.0371)	0.0571 (0.0377)	0.0730 (0.0355)	**0.0250** (0.0141)	0.0492 (0.0509)	0.0508 (0.0316)	0.0566 (0.0345)	0.0374 (0.0305)
	0.1		0.0507 (0.0307)	0.0536 (0.0242)	0.0649 (0.0297)	**0.0358** (0.0205)	0.0440 (0.0459)	0.0374 (0.0318)	0.0416 (0.0227)	0.0378 (0.0236)
	0.2		0.0399 (0.0207)	0.0476 (0.0230)	0.0514 (0.0322)	0.0620 (0.0548)	**0.0222** (0.0146)	0.0445 (0.0234)	0.0443 (0.0202)	0.0362 (0.0200)
	0.3		0.0452 (0.0179)	0.0442 (0.0164)	0.0630 (0.0142)	0.0411 (0.0368)	**0.0221** (0.0170)	0.0373 (0.0169)	0.0443 (0.0186)	0.0404 (0.0281)
	1	0.0533 (0.0110)								
student	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0242 (0.0499)
	0.05		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0040 (0.0126)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0504 (0.0491)
	0.1		**0.0000** (0.0000)	**0.0000** (0.0000)	0.1225 (0.1670)	0.1202 (0.1078)	0.1220 (0.1049)	0.0071 (0.0226)	**0.0000** (0.0000)	0.0358 (0.0351)
	0.2		0.1189 (0.0889)	0.1310 (0.1045)	0.1343 (0.0844)	0.1536 (0.1008)	0.1486 (0.0706)	0.1281 (0.0843)	**0.0993** (0.0771)	0.1643 (0.0967)
	0.3		0.1753 (0.0664)	0.1680 (0.0895)	0.1699 (0.1200)	0.1645 (0.0851)	**0.1394** (0.0689)	0.1454 (0.0633)	0.1730 (0.0944)	0.1695 (0.0574)
	1	0.1328 (0.0498)								
	Wins	0	**7**	0	0	2	**7**	2	2	0

Table 13. Equalized Odds Difference results (HGB). Bold values indicate the lowest value per row.

Dataset	Ratio	HGB	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	0.1147 (0.1161)	0.0427 (0.0879)	0.0431 (0.0547)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0438 (0.0518)
	0.05		**0.0771** (0.1157)	0.1061 (0.0923)	0.1029 (0.0500)	0.1218 (0.1137)	0.1069 (0.0669)	0.1570 (0.2060)	0.2284 (0.2303)	0.1211 (0.1301)
	0.1		0.0806 (0.0372)	0.1040 (0.0529)	0.1211 (0.0954)	0.1398 (0.1064)	0.1251 (0.0929)	0.1229 (0.0924)	0.1430 (0.1169)	**0.0668** (0.0373)
	0.2		0.1170 (0.0599)	0.1273 (0.0744)	0.1131 (0.0573)	0.1386 (0.0755)	0.1080 (0.0361)	0.1081 (0.0444)	**0.0972** (0.0506)	0.1299 (0.0528)
	0.3		0.1208 (0.0568)	0.1129 (0.0329)	0.1217 (0.0776)	0.1279 (0.0614)	0.1225 (0.0735)	**0.1089** (0.0409)	0.1238 (0.0773)	0.1337 (0.0735)
	1	0.1013 (0.0709)								
eap2024	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0491 (0.0899)
	0.05		**0.0000** (0.0000)	**0.0000** (0.0000)	0.2123 (0.2880)	0.1080 (0.2026)	0.0412 (0.1304)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.1159 (0.1441)
	0.1		0.2597 (0.1876)	0.2681 (0.1374)	0.2988 (0.1432)	0.2460 (0.1313)	0.3157 (0.1251)	0.2282 (0.1457)	0.2397 (0.0738)	**0.2191** (0.1138)
	0.2		0.2289 (0.1341)	0.2150 (0.0763)	0.2675 (0.0884)	0.2561 (0.0990)	**0.2112** (0.0746)	0.2586 (0.1020)	0.2163 (0.1211)	0.2162 (0.0931)
	0.3		0.2180 (0.1081)	0.2335 (0.0903)	0.2737 (0.1153)	**0.1652** (0.0630)	0.1982 (0.0810)	0.2454 (0.1134)	0.2275 (0.0953)	0.1985 (0.0753)
	1	0.2323 (0.1131)								
MBA	0.01		0.0764 (0.0470)	0.0673 (0.0593)	0.0948 (0.0613)	0.0908 (0.0425)	0.0971 (0.0676)	**0.0287** (0.0134)	0.0610 (0.0512)	0.0473 (0.0465)
	0.05		0.1159 (0.0798)	0.1209 (0.0747)	0.1092 (0.0586)	**0.0518** (0.0352)	0.1451 (0.0807)	0.1124 (0.0574)	0.1094 (0.0552)	0.0743 (0.0672)
	0.1		0.0842 (0.0480)	0.0904 (0.0463)	0.0994 (0.0512)	0.0845 (0.0368)	0.1102 (0.0712)	0.0713 (0.0390)	**0.0621** (0.0372)	0.0687 (0.0468)
	0.2		0.0903 (0.0397)	0.0800 (0.0503)	0.0873 (0.0428)	0.1512 (0.1104)	**0.0665** (0.0524)	0.0768 (0.0583)	0.0854 (0.0440)	0.0666 (0.0462)
	0.3		0.0683 (0.0390)	0.0669 (0.0444)	0.0969 (0.0731)	0.0906 (0.0571)	0.0591 (0.0352)	**0.0550** (0.0236)	0.0784 (0.0501)	0.0688 (0.0416)
	1	0.0808 (0.0406)								
student	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0486 (0.0912)
	0.05		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.0056 (0.0176)	**0.0000** (0.0000)	**0.0000** (0.0000)	0.1194 (0.1298)
	0.1		**0.0000** (0.0000)	**0.0000** (0.0000)	0.2052 (0.2466)	0.2714 (0.1982)	0.3046 (0.2014)	0.0200 (0.0632)	**0.0000** (0.0000)	0.1042 (0.0857)
	0.2		0.2901 (0.1244)	0.2288 (0.1581)	0.2624 (0.1546)	0.2501 (0.1680)	0.2362 (0.1648)	0.3043 (0.1702)	**0.1246** (0.0852)	0.2575 (0.1745)
	0.3		0.3209 (0.1949)	0.2581 (0.1514)	**0.2558** (0.1461)	0.2928 (0.1805)	0.1967 (0.0903)	0.2775 (0.1728)	0.2581 (0.1514)	0.2609 (0.1788)
	1	0.2130 (0.1592)								
	Wins	0	**7**	0	0	2	3	3	3	2

The accuracy results in Table 14 indicate that performance varies significantly across datasets, labeled ratios, and semi-supervised methods. For very low ratios (0.01), accuracy is generally low, especially in eap2024 and student, where models remain close to random guessing, while cert and MBA have good baseline performance even with minimal supervision. As the ratio increases, there are substantial improvements on all datasets, with cert achieving the highest overall accuracies (well above 0.93 for several methods at ratios ($\geq$0.1)), closely followed by MBA and student, with eap2024 consistently lagging behind. Method-wise, ensemble-based techniques such as Tri-Training and CoBC consistently achieve the best accuracy, as evident from their high wins count, whereas Self-Training and Setred have more variable benefits. Rasco and Rel-Rasco are competitive across a number of datasets, particularly for cert and student. These findings collectively imply that while semi-supervised learning can greatly boost predictive accuracy compared to very low-label regimes, relative improvements are dataset-dependent, and ensemble approaches will generally yield the highest accuracy.

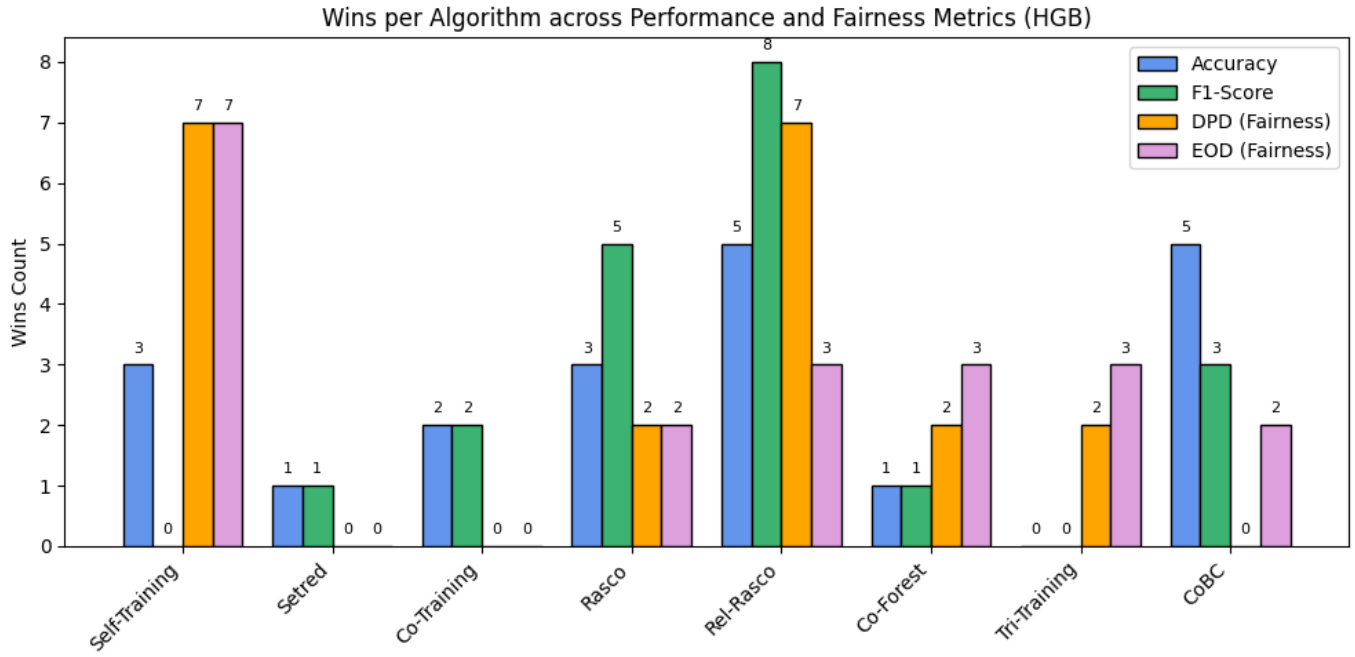

Figure 3. Summary of performance across SSL algorithms using HGB as the base classifier.

The F1-score values in Table 15 provide a more even view of model performance, indicating precision–recall trade-off across datasets and labeling proportions. F1-scores in the cert dataset are strong at medium labeling proportions ($\geq$0.1) where all approaches consistently rate > 0.90, with CoBC and Rasco often having the best result. At very low supervision (0.01), however, F1-scores drop dramatically (0.47–0.55), with no stable predictions. The eap2024 dataset gets better with the rising ratio, with values increasing from 0.34 at low labeling to 0.77–0.79 at rising ratios, with Rasco and Rel-Rasco being predominantly the top performers. MBA performance is comparatively low with F1-scores remaining at 0.37–0.43 even at rising ratios, suggesting that this dataset has slightly more difficulty in achieving balanced precision–recall performance. The student set shows one of the largest improvements, jumping from 0.34 at 0.01 to >0.82–0.85 at ($\geq$0.1) ratios, with Rel-Rasco and CoBC faring best. Overall, ensemble methods (Rasco and Rel-Rasco specifically) are the most competitive across sets, with the most wins, while simpler methods such as Self-Training and Setred are more dataset dependent.

Table 14. Accuracy results (XGB). Bold values indicate the highest value per row.

Dataset	Ratio	XGB	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.7412 (0.1009)	0.7412 (0.1009)	0.7937 (0.0611)	**0.8013** (0.0730)	0.7991 (0.0664)	0.7080 (0.1619)	0.7444 (0.1000)	0.7280 (0.1360)
	0.05		0.9156 (0.0285)	0.9188 (0.0242)	0.9295 (0.0242)	**0.9338** (0.0251)	0.9274 (0.0279)	0.8995 (0.0403)	0.9146 (0.0333)	0.9103 (0.0306)
	0.1		0.9327 (0.0202)	0.9338 (0.0198)	0.9380 (0.0180)	0.9391 (0.0196)	0.9391 (0.0207)	0.9264 (0.0355)	0.9381 (0.0233)	**0.9402** (0.0167)
	0.2		0.9284 (0.0281)	0.9241 (0.0280)	0.9369 (0.0148)	0.9348 (0.0178)	0.9273 (0.0208)	0.9263 (0.0239)	**0.9370** (0.0205)	0.9327 (0.0237)
	0.3		0.9252 (0.0202)	0.9252 (0.0150)	0.9338 (0.0149)	0.9305 (0.0228)	0.9306 (0.0161)	0.9380 (0.0230)	0.9305 (0.0198)	**0.9391** (0.0174)
	1	0.9380 (0.0181)								
eap2024	0.01		**0.5168** (0.0243)	**0.5168** (0.0243)	**0.5168** (0.0243)	**0.5168** (0.0243)	**0.5168** (0.0243)	**0.5168** (0.0243)	0.5102 (0.0281)	0.5044 (0.0287)
	0.05		0.7462 (0.0432)	0.7528 (0.0541)	0.7194 (0.0688)	0.7640 (0.0550)	0.7526 (0.0542)	0.7127 (0.0753)	**0.7662** (0.0537)	0.7575 (0.0470)
	0.1		0.7435 (0.0767)	0.7367 (0.0580)	0.7480 (0.0499)	0.7728 (0.0483)	**0.7795** (0.0349)	0.7322 (0.0646)	0.7481 (0.0664)	0.7482 (0.0548)
	0.2		0.7594 (0.0430)	0.7663 (0.0512)	0.7415 (0.0472)	0.7664 (0.0446)	0.7619 (0.0378)	0.7575 (0.0517)	**0.7685** (0.0543)	0.7663 (0.0467)
	0.3		0.7547 (0.0398)	0.7526 (0.0569)	0.7776 (0.0579)	**0.7888** (0.0347)	0.7776 (0.0368)	0.7839 (0.0588)	0.7638 (0.0355)	0.7729 (0.0436)
	1	0.7751 (0.0481)								
MBA	0.01		0.8077 (0.0179)	0.8063 (0.0175)	0.7472 (0.0252)	0.7906 (0.0264)	0.7904 (0.0345)	0.8034 (0.0257)	0.8098 (0.0201)	**0.8239** (0.0142)
	0.05		0.8156 (0.0137)	0.8074 (0.0129)	0.7522 (0.0217)	0.7911 (0.0134)	0.7845 (0.0187)	0.7982 (0.0241)	0.8118 (0.0112)	**0.8243** (0.0117)
	0.1		0.8109 (0.0127)	0.8008 (0.0168)	0.7698 (0.0192)	0.7964 (0.0180)	0.8027 (0.0193)	0.8092 (0.0115)	0.8114 (0.0134)	**0.8187** (0.0119)
	0.2		0.8208 (0.0109)	0.8035 (0.0112)	0.7901 (0.0140)	0.8066 (0.0111)	0.8037 (0.0121)	0.8116 (0.0141)	0.8116 (0.0129)	**0.8240** (0.0082)
	0.3		0.8174 (0.0091)	0.8119 (0.0081)	0.8019 (0.0104)	0.8140 (0.0137)	0.8140 (0.0124)	0.8158 (0.0086)	0.8160 (0.0081)	**0.8253** (0.0082)
	1	0.8210 (0.0151)								
student	0.01		**0.5380** (0.1485)	**0.5380** (0.1485)	**0.5380** (0.1485)	**0.5380** (0.1485)	**0.5380** (0.1485)	**0.5380** (0.1485)	0.4620 (0.1485)	**0.5380** (0.1485)
	0.05		0.7099 (0.1083)	0.6713 (0.1261)	0.6917 (0.1207)	0.7000 (0.1151)	0.7093 (0.1585)	0.7208 (0.1070)	**0.7471** (0.1376)	0.7241 (0.0718)
	0.1		0.8456 (0.0761)	0.8455 (0.0467)	0.7297 (0.0718)	0.7902 (0.0713)	**0.8545** (0.0570)	0.8487 (0.0907)	0.8517 (0.0560)	0.8544 (0.0875)
	0.2		0.8429 (0.0747)	0.8399 (0.0760)	0.7788 (0.0588)	0.8228 (0.0622)	0.8634 (0.0715)	0.8428 (0.0699)	0.8427 (0.0823)	**0.8661** (0.0651)
	0.3		0.8403 (0.0754)	0.8316 (0.0694)	0.8052 (0.0496)	0.8545 (0.0664)	**0.8634** (0.0769)	0.8574 (0.0773)	0.8345 (0.0628)	0.8574 (0.0848)
	1	0.8169 (0.0646)								
	Wins	0	2	0	0	3	3	0	4	**8**

Table 15. F1-Score results (XGB). Bold values indicate the highest value per row.

Dataset	Ratio	XGB	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.4708 (0.1214)	0.4708 (0.1214)	0.5531 (0.1968)	**0.5578** (0.2086)	0.5539 (0.1984)	0.4123 (0.0764)	0.4826 (0.1060)	0.5023 (0.1177)
	0.05		0.8813 (0.0520)	0.8865 (0.0365)	0.9019 (0.0378)	**0.9073** (0.0399)	0.8993 (0.0410)	0.8544 (0.0722)	0.8759 (0.0606)	0.8678 (0.0494)
	0.1		0.9103 (0.0266)	0.9109 (0.0253)	0.9180 (0.0225)	0.9184 (0.0268)	0.9182 (0.0273)	0.8918 (0.0746)	0.9172 (0.0290)	**0.9209** (0.0195)
	0.2		0.9033 (0.0383)	0.8977 (0.0394)	**0.9158** (0.0197)	0.9129 (0.0239)	0.9031 (0.0278)	0.9006 (0.0325)	0.9151 (0.0278)	0.9090 (0.0330)
	0.3		0.8983 (0.0284)	0.8971 (0.0243)	0.9110 (0.0205)	0.9070 (0.0316)	0.9063 (0.0235)	0.9152 (0.0324)	0.9059 (0.0291)	**0.9184** (0.0232)
	1	0.9156 (0.0275)								
eap2024	0.01		**0.3406** (0.0108)	**0.3406** (0.0108)	**0.3406** (0.0108)	**0.3406** (0.0108)	**0.3406** (0.0108)	**0.3406** (0.0108)	0.3376 (0.0125)	0.3351 (0.0128)
	0.05		0.7422 (0.0453)	0.7480 (0.0583)	0.7126 (0.0742)	0.7587 (0.0599)	0.7487 (0.0579)	0.7061 (0.0790)	**0.7607** (0.0580)	0.7515 (0.0480)
	0.1		0.7401 (0.0777)	0.7339 (0.0573)	0.7455 (0.0496)	0.7706 (0.0487)	**0.7786** (0.0350)	0.7270 (0.0686)	0.7452 (0.0666)	0.7452 (0.0548)
	0.2		0.7573 (0.0442)	0.7640 (0.0515)	0.7378 (0.0480)	0.7645 (0.0448)	0.7599 (0.0389)	0.7550 (0.0519)	**0.7665** (0.0561)	0.7652 (0.0469)
	0.3		0.7528 (0.0397)	0.7511 (0.0580)	0.7752 (0.0600)	**0.7875** (0.0356)	0.7766 (0.0372)	0.7816 (0.0594)	0.7627 (0.0358)	0.7711 (0.0443)
	1	0.7739 (0.0483)								
MBA	0.01		0.3662 (0.0377)	0.3750 (0.0355)	**0.3954** (0.0221)	**0.3954** (0.0325)	0.3878 (0.0269)	0.3742 (0.0406)	0.3710 (0.0365)	0.3523 (0.0343)
	0.05		0.3995 (0.0251)	0.4139 (0.0379)	0.3997 (0.0183)	**0.4280** (0.0257)	0.4178 (0.0428)	0.3865 (0.0234)	0.3980 (0.0255)	0.3832 (0.0251)
	0.1		0.3944 (0.0168)	0.4015 (0.0220)	0.4077 (0.0164)	0.4207 (0.0079)	**0.4291** (0.0214)	0.4062 (0.0293)	0.4031 (0.0329)	0.3781 (0.0182)
	0.2		0.4035 (0.0129)	0.4022 (0.0133)	0.4132 (0.0242)	**0.4227** (0.0111)	0.4106 (0.0206)	0.3914 (0.0090)	0.4014 (0.0145)	0.3906 (0.0158)
	0.3		0.3934 (0.0168)	0.4101 (0.0125)	0.4053 (0.0282)	**0.4167** (0.0156)	0.4138 (0.0184)	0.4027 (0.0191)	0.4082 (0.0075)	0.3926 (0.0169)
	1	0.4029 (0.0214)								
student	0.01		**0.3442** (0.0664)	**0.3442** (0.0664)	**0.3442** (0.0664)	**0.3442** (0.0664)	**0.3442** (0.0664)	**0.3442** (0.0664)	0.3101 (0.0668)	**0.3442** (0.0664)
	0.05		0.5993 (0.1916)	0.6076 (0.1679)	0.6401 (0.1577)	0.6540 (0.1507)	0.6657 (0.1842)	0.6514 (0.1590)	**0.7018** (0.1766)	0.6489 (0.1336)
	0.1		0.8263 (0.0858)	0.8200 (0.0603)	0.6791 (0.0913)	0.7592 (0.0865)	**0.8361** (0.0669)	0.8123 (0.1309)	0.8306 (0.0670)	0.8236 (0.1259)
	0.2		0.8249 (0.0824)	0.8205 (0.0877)	0.7497 (0.0668)	0.7972 (0.0761)	0.8472 (0.0806)	0.8246 (0.0818)	0.8241 (0.0929)	**0.8475** (0.0761)
	0.3		0.8243 (0.0832)	0.8146 (0.0751)	0.7776 (0.0645)	0.8338 (0.0761)	**0.8425** (0.0897)	0.8408 (0.0882)	0.8176 (0.0688)	0.8389 (0.0961)
	1	0.7933 (0.0742)								
	Wins	0	2	0	2	**6**	4	0	3	3

The Demographic Parality Difference (DPD) values in Table 16 indicate major trends regarding fairness between datasets, labeled ratios, and learning methods. For the cert dataset, the values are consistently low (0.01–0.11), indicating that all approaches make relatively balanced predictions within protected groups, whereas CoBC at times yields higher imbalances. In eap2024, fairness outcomes range extremely widely: no imbalance is observed at the lowest ratio (0.01), but values rise dramatically at higher ratios, usually far above 0.16, displaying rising bias with the addition of more labeled data. The MBA dataset shows the most uniform and weak disparities overall (0.02–0.07), suggesting low sensitivity to semi-supervised learning methods. On the other hand, the student dataset exhibits the most variability and the highest disparities, and values of DPD up to 0.23 at a ratio of 0.2 reflect more fairness problems. Method-wise, there is no single method that minimizes DPD for all datasets at once, though Co-Forest and Rasco show lower disparities in most cases, and sometimes CoBC and Co-Training bring in greater differences.

Equalized Odds Difference (EOD) figures in Table 17 reveal significant differences among datasets, suggesting that semi-supervised approaches can exaggerate or reduce fairness differences based on the properties of the data and labeling proportions. For cert data, EOD values are relatively small (0.05–0.14), and Co-Forest and Co-Training tend to produce lower differences, but variability increases with larger amounts of labeled data. In eap2024, the differences are small at 0.01 but increase rapidly with higher ratios, typically over 0.25–0.32, which implies that increased supervision can reinforce unfair treatment between groups; Rasco, Rel-Rasco, and Co-Training are likely to record the largest differences. The MBA dataset is more robust with values typically below 0.13 and less labeled ratio sensitivity, implying a relatively balanced group treatment across techniques. On the other hand, the student set displays the most challenging fairness profile: while no disparity exists for 0.01, it leaps to 0.25–0.37 at higher ratios, where Co-Training particularly reaches the peak disparities. From a methodological perspective, Self-Training and Rasco win the most (five for each), but none of the methods reduce EOD across all datasets uniformly.

As shown in Figure 4, certain semi-supervised methods outperform others when XGB is used as the base classifier, indicating differences in their adaptability to the datasets.

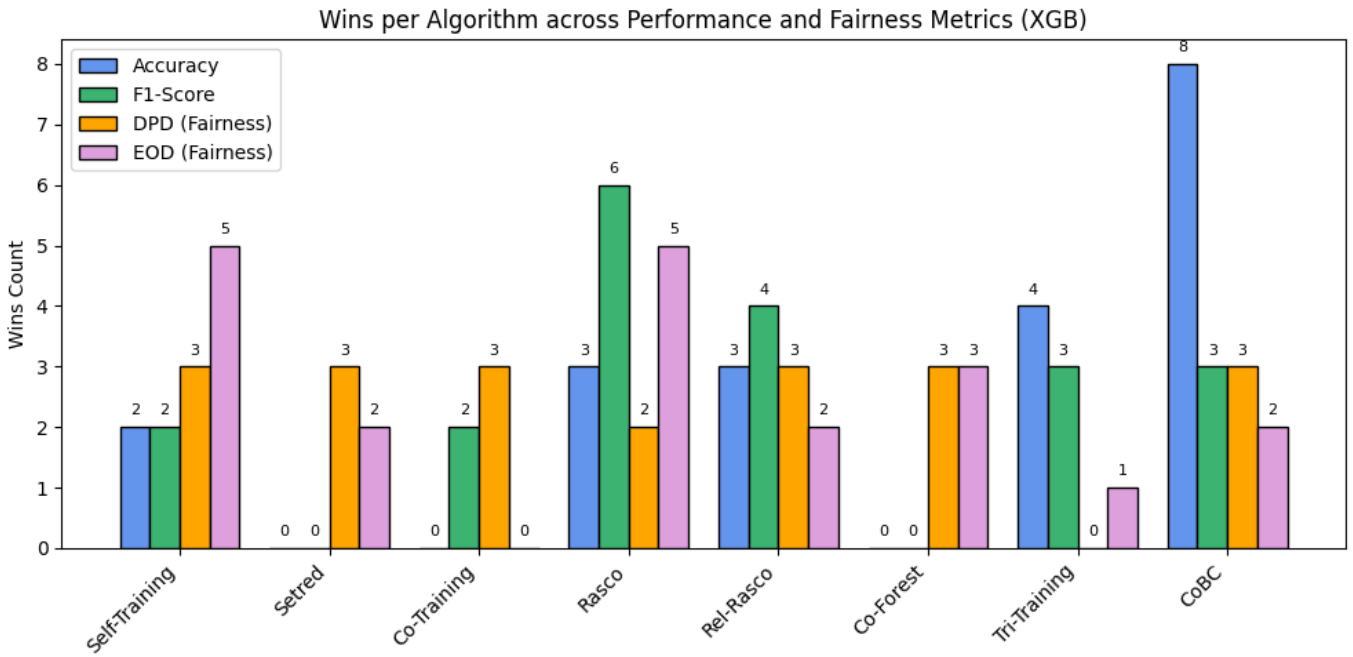

Figure 4. Summary of performance across SSL algorithms using XGB as the base classifier.

Table 16. Demographic Parity Difference results (XGB). Bold values indicate the lowest value per row.

Dataset	Ratio	XGB	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.0476 (0.1349)	0.0476 (0.1349)	0.0141 (0.0310)	0.0107 (0.0303)	0.0167 (0.0353)	**0.0028** (0.0088)	0.0551 (0.1341)	0.0786 (0.1506)
	0.05		0.0890 (0.0842)	0.0811 (0.0805)	0.0860 (0.0828)	0.0895 (0.0810)	0.0848 (0.0767)	**0.0810** (0.0853)	0.0854 (0.0810)	0.0918 (0.0666)
	0.1		0.0723 (0.0902)	0.0764 (0.1003)	0.0754 (0.0789)	0.0894 (0.0941)	0.0890 (0.0920)	**0.0678** (0.0708)	0.0735 (0.0911)	0.0916 (0.1061)
	0.2		0.1007 (0.0721)	**0.0867** (0.0709)	0.0938 (0.0845)	0.0931 (0.0802)	0.0937 (0.0737)	0.0984 (0.0901)	0.0968 (0.0873)	0.1071 (0.0879)
	0.3		0.0908 (0.0776)	**0.0744** (0.0580)	0.0890 (0.1026)	0.0906 (0.0913)	0.0948 (0.0822)	0.0998 (0.1035)	0.0847 (0.0871)	0.0939 (0.0976)
	1	0.0821 (0.0742)								
eap2024	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)
	0.05		0.1372 (0.1020)	**0.0979** (0.0742)	0.1131 (0.0817)	0.1249 (0.1002)	0.1447 (0.0437)	0.1588 (0.1052)	0.1145 (0.0645)	0.1525 (0.1019)
	0.1		0.1207 (0.0584)	0.0991 (0.0717)	**0.0863** (0.0764)	0.1295 (0.0840)	0.1217 (0.1194)	0.1067 (0.0592)	0.1168 (0.0865)	0.1430 (0.1000)
	0.2		0.1605 (0.1066)	0.1470 (0.1109)	0.1408 (0.0574)	0.1148 (0.0840)	0.1633 (0.0783)	0.1239 (0.0932)	0.1233 (0.0803)	**0.0847** (0.0589)
	0.3		**0.0971** (0.0879)	0.1146 (0.0938)	0.1750 (0.0986)	0.1253 (0.0861)	0.1189 (0.0734)	0.1565 (0.1156)	0.1005 (0.0772)	0.1393 (0.1171)
	1	0.1707 (0.0697)								
MBA	0.01		0.0235 (0.0144)	0.0310 (0.0174)	0.0357 (0.0304)	0.0200 (0.0172)	0.0321 (0.0311)	0.0207 (0.0180)	0.0222 (0.0154)	**0.0073** (0.0074)
	0.05		0.0569 (0.0385)	0.0615 (0.0375)	0.0703 (0.0489)	0.0465 (0.0307)	**0.0309** (0.0282)	0.0475 (0.0257)	0.0575 (0.0397)	0.0478 (0.0513)
	0.1		0.0460 (0.0240)	0.0531 (0.0178)	0.0506 (0.0320)	0.0398 (0.0187)	**0.0382** (0.0403)	0.0510 (0.0317)	0.0443 (0.0203)	0.0387 (0.0338)
	0.2		0.0489 (0.0207)	0.0469 (0.0188)	0.0664 (0.0336)	0.0364 (0.0173)	**0.0207** (0.0266)	0.0411 (0.0213)	0.0513 (0.0208)	0.0451 (0.0232)
	0.3		0.0448 (0.0183)	0.0471 (0.0129)	0.0615 (0.0247)	**0.0304** (0.0210)	0.0311 (0.0329)	0.0400 (0.0225)	0.0456 (0.0170)	0.0525 (0.0239)
	1	0.0492 (0.0179)								
student	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)
	0.05		0.1245 (0.1177)	0.1304 (0.1573)	0.1987 (0.1794)	0.1559 (0.1570)	0.1468 (0.1014)	0.1173 (0.1758)	0.1416 (0.1328)	**0.0994** (0.1209)
	0.1		0.1855 (0.0836)	0.1592 (0.1000)	**0.1441** (0.1430)	0.1500 (0.0975)	0.1550 (0.0921)	0.1602 (0.0816)	0.1867 (0.0902)	0.1478 (0.0722)
	0.2		0.1794 (0.0905)	0.1768 (0.1079)	**0.1563** (0.0802)	0.2337 (0.1544)	0.1742 (0.1190)	0.1658 (0.0882)	0.1952 (0.1071)	0.1626 (0.0921)
	0.3		0.1872 (0.1205)	0.1842 (0.1361)	0.1532 (0.0885)	**0.1460** (0.0942)	0.1888 (0.0740)	0.1616 (0.0984)	0.1775 (0.1238)	0.1837 (0.0820)
	1	0.1545 (0.0933)								
	Wins	0	3	3	3	2	3	3	0	3

Table 17. Equalized Odds Difference results (XGB). Bold values indicate the lowest value per row.

Dataset	Ratio	XGB	Self-Training	Setred	Co-Training	Rasco	Rel-Rasco	Co-Forest	Tri-Training	CoBC
cert	0.01		0.0499 (0.1496)	0.0499 (0.1496)	0.0219 (0.0402)	0.0456 (0.1090)	0.0566 (0.1468)	**0.0100** (0.0316)	0.0813 (0.1593)	0.1113 (0.1590)
	0.05		0.1202 (0.1073)	0.1441 (0.1132)	0.1132 (0.0608)	**0.1111** (0.0690)	0.1381 (0.0594)	0.1647 (0.1131)	0.1322 (0.0980)	0.1813 (0.0954)
	0.1		**0.0828** (0.0425)	0.0957 (0.0779)	0.0990 (0.0760)	0.1032 (0.0509)	0.0953 (0.0901)	0.1138 (0.0807)	0.0923 (0.0426)	0.0831 (0.0793)
	0.2		**0.0992** (0.0716)	0.1038 (0.0649)	0.1133 (0.0866)	0.1244 (0.0364)	0.1151 (0.0654)	0.1458 (0.0776)	0.1120 (0.0573)	0.1387 (0.0885)
	0.3		0.1407 (0.0837)	0.1132 (0.0853)	0.1325 (0.0796)	0.1104 (0.0540)	0.1080 (0.0579)	0.1427 (0.0905)	**0.1050** (0.0283)	0.1386 (0.0868)
	1	0.0977 (0.0529)								
eap2024	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)
	0.05		0.2480 (0.1161)	**0.1654** (0.1159)	0.2301 (0.0821)	0.2619 (0.1042)	0.2888 (0.1262)	0.2396 (0.1469)	0.2150 (0.0925)	0.2436 (0.1100)
	0.1		**0.2297** (0.1071)	0.2704 (0.0909)	0.2846 (0.1077)	0.2849 (0.0606)	0.2664 (0.1024)	0.2748 (0.1547)	0.2445 (0.1157)	0.2858 (0.0642)
	0.2		0.2650 (0.1437)	0.2750 (0.1195)	0.3253 (0.1045)	0.2801 (0.0900)	0.2709 (0.0917)	**0.2235** (0.0829)	0.2660 (0.1069)	0.2472 (0.1112)
	0.3		0.2536 (0.1276)	0.2678 (0.1239)	0.2532 (0.1239)	0.2570 (0.0963)	0.2593 (0.1158)	0.2363 (0.1285)	0.2345 (0.1178)	**0.2196** (0.1163)
	1	0.3028 (0.0867)								
MBA	0.01		0.0413 (0.0238)	0.0566 (0.0235)	0.0680 (0.0409)	0.0571 (0.0309)	0.0849 (0.0451)	0.0408 (0.0241)	0.0401 (0.0230)	**0.0234** (0.0233)
	0.05		0.1197 (0.0813)	0.1247 (0.0871)	0.1188 (0.0523)	0.1243 (0.0474)	0.1140 (0.0632)	**0.1048** (0.0683)	0.1166 (0.0651)	0.1151 (0.0892)
	0.1		0.0804 (0.0458)	0.0867 (0.0538)	0.0823 (0.0494)	**0.0747** (0.0354)	0.1332 (0.1017)	0.0960 (0.0514)	0.0913 (0.0531)	0.0763 (0.0579)
	0.2		0.0930 (0.0559)	**0.0661** (0.0464)	0.0911 (0.0707)	0.0839 (0.0479)	0.0810 (0.0545)	0.0725 (0.0657)	0.0768 (0.0541)	0.0922 (0.0638)
	0.3		0.0727 (0.0288)	0.0721 (0.0257)	0.0964 (0.0622)	0.0812 (0.0556)	**0.0664** (0.0348)	0.0681 (0.0301)	0.0746 (0.0279)	0.0872 (0.0416)
	1	0.0777 (0.0508)								
student	0.01		**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)	**0.0000** (0.0000)
	0.05		0.2887 (0.2788)	0.2662 (0.2735)	0.3716 (0.3193)	**0.1809** (0.2003)	0.2916 (0.2013)	0.2198 (0.2092)	0.2423 (0.2479)	0.2281 (0.1739)
	0.1		0.2673 (0.1705)	0.2915 (0.1627)	0.3287 (0.2028)	**0.2009** (0.1376)	0.2152 (0.1278)	0.2236 (0.1494)	0.3133 (0.1683)	0.2566 (0.1415)
	0.2		0.2607 (0.1515)	0.2630 (0.1715)	0.3372 (0.1542)	0.3277 (0.2035)	**0.2296** (0.1421)	0.2829 (0.2111)	0.2738 (0.2155)	0.3018 (0.1748)
	0.3		0.3047 (0.2143)	0.2936 (0.2096)	0.3240 (0.1250)	**0.2490** (0.2062)	0.2982 (0.1979)	0.2675 (0.1563)	0.3010 (0.2038)	0.2717 (0.1998)
	1	0.2677 (0.1695)								
	Wins	0	**5**	2	0	**5**	2	3	1	2

6. Discussion

The experimental results on four diverse datasets demonstrate the competitive advantage of SSL methods over fully supervised baselines, particularly in low-label regimes. Comparisons of accuracy and F1-score consistently demonstrate that methods like CoBC, Rasco, and Rel-Rasco are superior or on par with fully supervised methods like RF, HGB, or XGB even with only 1–5% of the labeled examples. CoBC tends to be the top in accuracy, while Rel-Rasco is top in F1-score rankings on noisy or imbalanced datasets like eap2024 and student. Simpler methods like Self-Training are also effective across settings, especially for MBA, and thus show their practicability. As the labeled ratio increases, all the methods move towards comparable performance, suggesting that SSL methods are most beneficial when data is limited. Notably, baselines under 100% supervision seldom outperform top SSL methods, even at 100% labeling, as well, underlining the value of SSL under real-world scenarios where labeled data is expensive or scarce.

Ensemble and multi-view algorithms like Rasco and CoBC perform better in accuracy compared to less complex methods like Self-Training due to their ability to exploit diversity in learners and feature representations. By having several classifiers trained on various subspaces or views of the data, these algorithms minimize overfitting risk and counter labeling error propagation, a typical drawback of single-view self-labeling. In the SSL settings with limited labeled data, this heterogeneity causes resilience so that the model will be capable of unveiling complementary patterns across views that could be obscured to an individual classifier. Hence, ensemble and multi-view methods capitalize on the structure of the data and the redundancy among the sets of features, rendering the predictions more stable and accurate, especially with complex or imbalanced data.

Apart from predictive performance, fairness metrics exhibit extreme differences across SSL methods, on both Demographic Parity Difference (DPD) and Equalized Odds Difference (EOD). Self-Training and Rel-Rasco always produce the smallest disparities, suggesting that these methods better minimize bias across demographic groups.

Notably, Self-Training achieves high performance on fairness metrics in spite of its overall lower accuracy. A likely explanation is that the less complex process of its mechanism, which consists of adding the most confidently predicted labels iteratively, is less damaging in making use of weak correlations in the data that could represent demographic bias. More sophisticated methods, while better at extracting predictive signal, may indeed also capture and amplify spurious correlations with sensitive features. Lower complexity and more cautious labeling of Self-Training, however, may disavow reinforcing its bias patterns from the small labeled set disproportionately. This lines up with the idea that fairness and accuracy do not always go hand in hand and that, sometimes, less sophisticated methods produce more fair results.

CoBC is also well-ranked on many fairness metrics, whereas ensemble-based methods like Co-Forest, Co-Training, and Tri-Training are more likely to generate more disparities, particularly on sensitive datasets such as eap2024 and student. Surprisingly, fairness is not necessarily improved by more abundant labeled data; occasionally DPD and EOD values increase with label richness, potentially through extrapolation of inherent bias in training data. Fully supervised baselines such as HGB and XGB also fail to achieve competitive fairness, especially on more challenging datasets, proving that label sufficiency alone cannot guarantee fairness. This suggests that optimal performance for accuracy does not necessarily equal optimal performance for fairness, and in some cases, the introduction of more labeled data actually made fairness worse by scaling up built-in bias in the data. In education, where predictive models can influence interventions and resource allocation, these results serve to emphasize the need to balance predictive performance with fairness, as well as to be cognizant of fairness-aware variants of SSL methods. These findings

justify fairness-aware SSL design for fair and effective machine learning in real-world, imbalanced settings.

It can be contributed to the bias amplification problem of SSL that fairness metrics sometimes get worse as more labeled data are added. At low data availability of labels, SSL algorithms rely a lot on the geometry of unlabeled data and synthetic augmentation, which assists in countering some biases in the sparse training set to a certain extent. But with more labeled data, models come to rely more on the labels, so that any initial demographic or systemic bias in the labeled set becomes amplified. This results in worse disparities in demographic parity and equalized odds measures. In essence, more labels make the model more certain to replicate biased patterns, which appears as deteriorating fairness scores, even with improvement in predictive performance.

These observations highlight the trade-offs between fairness and accuracy in SSL. Methods such as Rasco and CoBC generally perform very well on accuracy but are likely to show variation in fairness measures, especially on sensitive datasets such as student and eap2024. Methods such as Self-Training are less accurate but tend to produce fairer outcomes, mainly in terms of demographic parity. This suggests that the most accurate method is not the best in practice if fairness is also a key concern. The compromise reflects the value of hybrid methods or fairness-aware tweaks that weigh predictive ability against equity, so that improving performance is not achieved by systematically discriminating against some student groups. Figure 5 illustrates how the relative success of each semi-supervised algorithm varies across evaluation metrics and base classifiers.

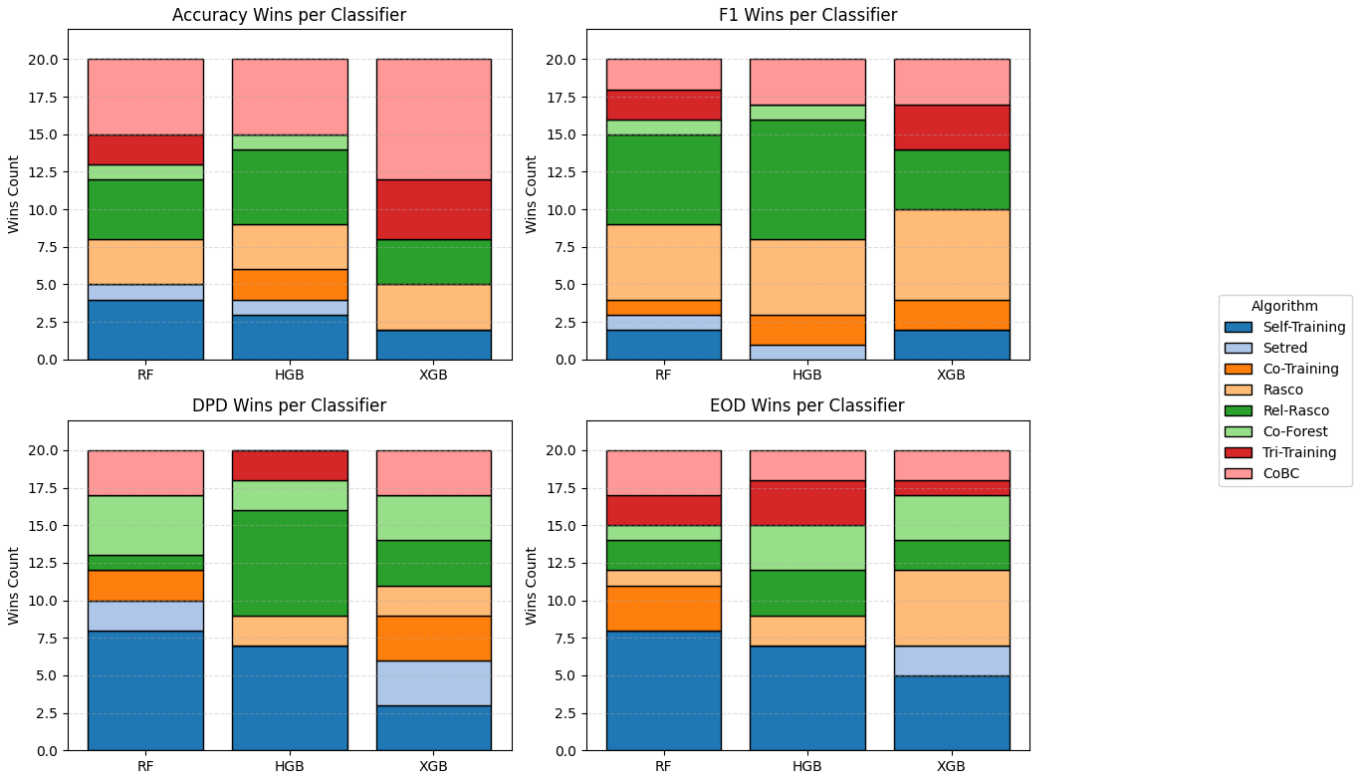

Figure 5. Stacked wins per evaluation metric across base classifiers.

We performed a direct comparison between baseline SSL methods and their TVAE-SSL augmented counterparts (Table 18). The results demonstrate that the method proposed was indeed tested and, while the size of improvement varied by dataset and method, TVAE augmentation consistently demonstrated large gains in many cases. For example, Rasco (TVAE-SSL) performs better than base Rasco on cert (0.9154 vs. 0.9027), eap2024 (0.6903 vs. 0.6817), and student (0.7220 vs. 0.7086). Similarly, Rel-Rasco (TVAE-SSL) performs better on

cert (0.9179 vs. 0.8985) and maintains as well as ever on student. Co-Training also benefits from TVAE augmentation on cert (0.9032 vs. 0.9000) and student (0.6895 vs. 0.6769). While in certain cases the differences are marginal or level (e.g., Setred, Co-Forest), the general trend seems to be that TVAE-SSL offers consistent or superior performance without degradation. We chose to report on HistGradientBoosting (HGB) compared to XGBoost (XGB) or Random Forests (RFs) as HGB always provided a reasonable trade-off between accuracy, justice, and computation cost across datasets. While RF overall produced a good solid baseline, in most cases its performance was surpassed by boosting-based methods, and XGB, although very accurate, tended to inflate fairness gaps and was computationally more expensive, especially in iterated semi-supervised runs with synthetic growth. HGB, on the other hand, posted results that were both consistent and robust across measures, thus serving as a representative option for illustrating the strengths of our TVAE-SSL framework without burdening the reader with duplicate comparisons.

Table 18. Average accuracy across all labeled ratios (HGB).

Algorithm	Cert	eap2024	MBA	Student
Self-Training (TVAE-SSL)	0.8705 (0.0773)	0.6607 (0.1347)	0.8120 (0.0066)	0.6987 (0.1296)
Self-Training (SSL)	0.8732 (0.0757)	0.6602 (0.1341)	0.8140 (0.0058)	0.7005 (0.1316)
Setred (TVAE-SSL)	0.8630 (0.0875)	0.6670 (0.1390)	0.8050 (0.0042)	0.6918 (0.1222)
Setred (SSL)	0.8630 (0.0875)	0.6670 (0.1390)	0.8050 (0.0042)	0.6918 (0.1222)
Co-Training (TVAE-SSL)	0.9032 (0.0657)	0.6751 (0.1176)	0.7693 (0.0212)	0.6895 (0.1056)
Co-Training (SSL)	0.9000 (0.0686)	0.6643 (0.1300)	0.7869 (0.0169)	0.6769 (0.0943)
Rasco (TVAE-SSL)	0.9154 (0.0372)	**0.6903** (0.1413)	0.7955 (0.0174)	0.7220 (0.1308)
Rasco (SSL)	0.9027 (0.0699)	0.6817 (0.1275)	0.8021 (0.0163)	0.7086 (0.1183)
Rel-Rasco (TVAE-SSL)	**0.9179** (0.0344)	0.6782 (0.1426)	0.7929 (0.0210)	**0.7318** (0.1432)
Rel-Rasco (SSL)	0.8985 (0.0777)	0.6835 (0.1239)	0.8003 (0.0136)	0.7289 (0.1398)
Co-Forest (TVAE-SSL)	0.8572 (0.0937)	0.6533 (0.1346)	0.8111 (0.0078)	0.6743 (0.1405)
Co-Forest (SSL)	0.8574 (0.0945)	0.6547 (0.1350)	0.8108 (0.0080)	0.6724 (0.1392)
Tri-Training (TVAE-SSL)	0.8769 (0.0778)	0.6741 (0.1510)	0.8103 (0.0017)	0.6836 (0.1441)
Tri-Training (SSL)	0.8790 (0.0760)	0.6727 (0.1498)	0.8119 (0.0038)	0.6877 (0.1470)
CoBC (TVAE-SSL)	0.8756 (0.0792)	0.6828 (0.1370)	**0.8254** (0.0031)	0.7015 (0.1322)
CoBC (SSL)	0.8771 (0.0776)	0.6652 (0.1277)	0.8229 (0.0030)	0.7004 (0.1322)

While overall improvements were brought about by SSL with TVAE augmentation, anomalies and shortcomings were observed. For some datasets, primarily eap2024 and student, performance gains were restricted or fairness scores deteriorated at higher labeled ratios, indicating that data quality and inherent bias may constrain SSL benefits. Similarly, MBA provided strong accuracy but persistently poor F1-scores, which indicated extreme class unbalance that SSL could not fully mitigate. These anomalies underscore the dependence of SSL methods on dataset properties like label distribution and sensitive attribute interdependencies. The second limitation is that fairness outcomes were highly context-sensitive; certain methods that bridged demographic gaps did not generalize under equalized odds, illustrating the multifaceted nature of fairness. Hybrid approaches integrating generative augmentation with fairness-aware objectives must thus be explored in future, and studies need to be carried out on more varied and realistic educational settings.

7. Conclusions

In this paper, we introduced TVAE-SSL, a novel framework that exploits both the generative capabilities of the Tabular Variational Autoencoder (TVAE) and the flexibility of SSL methods to boost model performance under low-data regimes. By integrating the unlabeled dataset with realistic, label-free synthetic samples generated using TVAE,

the method augments the quantity as well as the diversity of the training data without introducing label noise. This generative augmentation technique enables traditional SSL algorithms, e.g., Self-Training, Co-Training, and ensemble methods, to generalize more effectively even when the number of labeled examples is extremely low.

Our extensive empirical evaluation on four diverse tabular datasets demonstrates that TVAE-SSL improves predictive accuracy and F1-score, even under low-label scenarios (1–10% labeled data). Among the methods tried, Rasco-based versions and CoBC worked best, and our proposed augmentation method improved their performance even further. Notably, the results also show that TVAE-SSL improves fairness metrics such as Demographic Parity and Equalized Odds, indicating that synthetic data can help not only accuracy but also fair model conduct. These findings justify the broader use of generative models as a valuable component of semi-supervised pipelines for tabular data, where data annotation is costly or impossible.

One of the primary constraints of this study is its scope, which is confined to four chosen educational datasets, a narrow set of base classifiers (RF, HGB, and XGB), and a single generative model, the Tabular Variational Autoencoder (TVAE). While such choices allow controlled and systematic analysis, they restrict the generalizability of the results. Different datasets, particularly those with other domains or distributions of sensitive features, can lead to different performance and fairness. Likewise, other generative approaches such as GAN-based or diffusion-based can have potentially more effective synthetic augmentation impacts, and other base classifiers can react differently to SSL algorithms.

Looking ahead, TVAE-SSL suggests several directions. First, the integration of uncertainty-aware sample filtering or confidence weighted synthetic instance selection has the potential to further improve performance. Second, generalizing the framework to multiclass or multilabel problems, as well as exploring its integration with other fairness-aware learning objectives, can increase its applicability. Finally, applying TVAE-SSL to real-world applications such as healthcare, finance, and cybersecurity, where labeled data are limited and fairness is critical, can validate its real-world efficacy. Overall, this work illustrates the potential of a union between generative modeling and SSL as a path towards more effective, data efficient, and fairer machine learning systems.

Author Contributions: Conceptualization, S.K.; methodology, N.F.; validation, N.F.; writing—original draft preparation, G.K.; writing—review and editing, S.K. and Y.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the European Union through the Competitiveness Programme (ESPA 2021–2027) under the project easyHPC@eco.plastics.industry (MIS: 6001593).

Data Availability Statement: The "MBA Admission Dataset" (Class 2025) on Kaggle provides a publicly accessible (https://www.kaggle.com/datasets/taweilo/mba-admission-dataset, accessed on 20 September 2025) and well-structured collection of application-related data. The Hellenic Open University Learning Analytics Datasets analyzed during the current study is available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest

References

1. Mikre, F. The roles of information communication technologies in education: Review article with emphasis to the computer and internet. *Ethiop. J. Educ. Sci.* **2011**, *6*, 109–126.
2. Romero, C.; Ventura, S. Educational data mining: A review of the state of the art. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **2010**, *40*, 601–618. [CrossRef]
3. Romero, C.; Ventura, S. Data mining in education. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2013**, *3*, 12–27. [CrossRef]
4. Romero, C.; Ventura, S. Educational data mining and learning analytics: An updated survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2020**, *10*, e1355. [CrossRef]

5. Siemens, G.; Baker, R.S.J.d. Learning analytics and educational data mining: Towards communication and collaboration. In Proceedings of the 2nd International Conference on Learning Analytics and Knowledge, Vancouver, BC, Canada, 29 April–2 May 2012 ; pp. 252–254.
6. Zhu, X.; Goldberg, A. *Introduction to Semi-Supervised Learning*; Morgan & Claypool Publishers: San Rafael, CA, USA, 2009.
7. Zhou, Z.-H. A brief introduction to weakly supervised learning. *Natl. Sci. Rev.* 2018, 5, 44–53. [CrossRef]
8. Chapelle, O.; Scholkopf, B.; Zien, A. Semi-supervised learning. *IEEE Trans. Neural Netw.* **2009**, *20*, 542. [CrossRef]
9. Kostopoulos, G.; Karlos, S.; Kotsiantis, S. Multiview learning for early prognosis of academic performance: A case study. *IEEE Trans. Learn. Technol.* **2019**, *12*, 212–224. [CrossRef]
10. Kostopoulos, G.; Kotsiantis, S. Exploiting semi-supervised learning in the education field: A critical survey. In *Advances in Machine Learning/Deep Learning-Based Technologies*; Springer: Cham, Switzerland, 2021; Volume 2, pp. 79–94.
11. Kitto, K.; Knight, S. Practical ethics for building learning analytics. *Br. J. Educ. Technol.* **2019**, *50*, 2855–2870. [CrossRef]
12. Moreno-Marcos, P.M.; Alario-Hoyos, C.; Muñoz-Merino, P.J.M.; Kloos, C.D. Prediction in MOOCs: A review and future research directions. *IEEE Trans. Learn. Technol.* **2018**, *12*, 384–401. [CrossRef]
13. Kostopoulos, G.; Kotsiantis, S.; Pintelas, P. Predicting student performance in distance higher education using semi-supervised techniques. In Proceedings of the Model and Data Engineering: 5th International Conference, MEDI 2015, Rhodes, Greece, 26–28 September 2015; pp. 259–270.
14. Kostopoulos, G.; Livieris, I.E.; Kotsiantis, S.; Tampakas, V. Enhancing high school students' performance based on semi-supervised methods. In Proceedings of the 2017 8th International Conference on Information, Intelligence, Systems & Applications (IISA), Larnaca, Cyprus, 28–30 August 2017; pp. 1–6.
15. Kostopoulos, G.; Kotsiantis, S.; Pintelas, P. Estimating student dropout in distance higher education using semi-supervised techniques. In Proceedings of the 19th Panhellenic Conference on Informatics, Athens, Greece, 1–3 October 2015; pp. 38–43.
16. Li, W.; Gao, M.; Li, H.; Xiong, Q.; Wen, J.; Wu, Z. Dropout prediction in MOOCs using behavior features and multi-view semi-supervised learning. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016; pp. 3130–3137.
17. Kostopoulos, G.; Kotsiantis, S.; Fazakis, N.; Koutsonikos, G.; Pierrakeas, C. A semi-supervised regression algorithm for grade prediction of students in distance learning courses. *Int. J. Artif. Intell. Tools* **2019**, *28*, 1940001. [CrossRef]
18. Karlos, S.; Kostopoulos, G.; Kotsiantis, S. Predicting and interpreting students' grades in distance higher education through a semi-regression method. *Appl. Sci.* **2020**, *10*, 8413. [CrossRef]
19. Lu, Y.; Shen, M.; Wang, H.; Wang, X.; van Rechem, C.; Fu, T.; Wei, W. Machine learning for synthetic data generation: A review. *arXiv* **2023**, arXiv:2302.04062.
20. Emam, K.E.; Mosquera, L.; Hoptroff, R. *Practical Synthetic Data Generation: Balancing Privacy and the Broad Availability of Data*; O'Reilly Media: Sebastopol, UK, 2020.
21. Nikolenko, S.I. *Synthetic Data for Deep Learning*; Springer: Berlin/Heidelberg, Germany, 2021.
22. Mehrabi, N.; Morstatter, F.; Saxena, N.; Lerman, K.; Galstyan, A. A survey on bias and fairness in machine learning. *ACM Comput. Surv. CSUR* **2021**, *54*, 115. [CrossRef]
23. James, S.; Harbron, C.; Branson, J.; Sundler, M. Synthetic data use: Exploring use cases to optimise data utility. *Discov. Artif. Intell.* **2021**, *1*, 15. [CrossRef]
24. Flanagan, B.; Majumdar, R.; Ogata, H. Fine grain synthetic educational data: Challenges and limitations of collaborative learning analytics. *IEEE Access* **2022**, *10*, 26230–26241. [CrossRef]
25. Hu, Q.; Yuille, A.; Zhou, Z. Synthetic data as validation. *arXiv* **2023**, arXiv:2310.16052. [CrossRef]
26. Chan, Y.; Pu, G.; Shanker, A.; Suresh, P.; Jenks, P.; Heyer, J.; Denton, S. Balancing cost and effectiveness of synthetic data generation strategies for llms. *arXiv* **2024**, arXiv:2409.19759. [CrossRef]
27. Binns, R. Fairness in machine learning: Lessons from political philosophy. In Proceedings of the Conference on Fairness, Accountability and Transparency, New York, NY, USA, 23–24 February 2018; pp. 149–159.
28. Holstein, K.; Vaughan, J.W.; Daumé, H., III; Dudik, M.; Wallach, H. Improving fairness in machine learning systems: What do industry practitioners need? In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, Glasgow, UK, 4–9 May 2019; pp. 1–16.
29. Friedler, S.A.; Scheidegger, C.; Venkatasubramanian, S.; Choudhary, S.; Hamilton, E.P.; Roth, D. A comparative study of fairness-enhancing interventions in machine learning. In Proceedings of the Conference on Fairness, Accountability, and Transparency, Atlanta, GA, USA, 29–31 January 2019; pp. 329–338.
30. Rahmani, A.M.; Groot, W.; Rahmani, H. Dropout in online higher education: A systematic literature review. *Int. J. Educ. Technol. High. Educ.* **2024**, *21*, 19. [CrossRef]
31. Ghasempour, S.; Esmaeeli, M.; Abbasi, A.; Hosseinzadeh, A.; Ebrahimi, H. Relationship between academic success, distance education learning environments, and its related factors among medical sciences students: A cross-sectional study. *BMC Med Educ.* **2023**, *23*, 847. [CrossRef]

32. Kostopoulos, G.; Panagiotakopoulos, T.; Kotsiantis, S.; Pierrakeas, C.; Kameas, A. Interpretable models for early prediction of certification in MOOCs: A case study on a MOOC for smart city professionals. *IEEE Access* **2021**, *9*, 165881–165891. [CrossRef]
33. Triguero, I.; García, S.; Herrera, F. Self-labeled techniques for semi-supervised learning: Taxonomy, software and empirical study. *Knowl. Inf. Syst.* **2015**, *42*, 245–284. [CrossRef]
34. Li, M.; Zhou, Z.-H. SETRED: Self-training with editing. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Hanoi, Vietnam, 18–20 May 2005; pp. 611–621.
35. Blum, A.; Mitchell, T. Combining labeled and unlabeled data with co-training. In Proceedings of the Eleventh Annual Conference on Computational Learning Theory, Madison, WI, USA, 24–26 July 1998; pp. 92–100.
36. Zhou, Z.-H.; Li, M. Semi-supervised learning by disagreement. *Knowl. Inf. Syst.* **2010**, *24*, 415–439. [CrossRef]
37. Li, M.; Zhou, Z.-H. Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples. *IEEE Trans. Syst. Man, Cybern.-Part A Syst. Hum.* **2007**, *37*, 1088–1098. [CrossRef]
38. Hady, M.F.A.; Schwenker, F. Co-training by committee: A new semi-supervised learning framework. In Proceedings of the 2008 IEEE International Conference on Data Mining Workshops, Pisa, Italy, 15–19 December 2008; pp. 563–572.
39. Wang, J.; Luo, S.; Zeng, X.-H. A random subspace method for co-training. In Proceedings of the 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), Hong Kong, China, 1–6 June 2008; pp. 195–200.
40. Yaslan, Y.; Cataltepe, Z. Co-training with relevant random subspaces. *Neurocomputing* **2010**, *73*, 1652–1661. [CrossRef]
41. Zhou, Z.-H.; Li, M. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 1529–1541. [CrossRef]
42. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
43. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
44. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–9.
45. Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; Zemel, R. Fairness through awareness. In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, Cambridge, MA, USA, 8–10 January 2012; pp. 214–226.
46. Hardt, M.; Price, E.; Srebro, N. Equality of opportunity in supervised learning. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 1–22.
47. Xu, L.; Skoularidou, M.; Cuesta-Infante, A.; Veeramachaneni, K. Modeling tabular data using conditional gan. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 1–15.