UNIVERSITÀ DEGLI STUDI DI
NAPOLI FEDERICO II

Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Tesi di Laurea Magistrale in Distributed Systems

Joint assessment of accuracy, fairness and privacy in Machine Learning systems

Anno Accademico 2023/2024

Relatori
Ch.mo prof. Stefano Russo
Ch.mo prof. Roberto Pietrantuono

Correlatore
Ing. Luca Giamattei

Candidato
Iole Morabito
matr. M63001448

Χαλεπὰ τὰ καλά

Plato, Republic 4.435c

Abstract

The role of automated decision making systems, based on Machine Learning (ML) algorithm, is becoming increasingly significant in a wide range of domains, including healthcare, law, economics, and finance.

As these subject areas are sensitive and impact human lives, it is not sufficient to assess system performance; there are also sensitive data and ethical aspects that cannot be overlooked. Consequently, it is important to evaluate fairness and privacy as quality targets not in isolation but in relation to each other.

Despite previous studies that have conducted trade-off analyses on accuracy-fairness or fairness-privacy, there is still a debate as to how these should be considered about one another. There is still uncertainty as to whether they are fully conflated or not.

This thesis aims to examine accuracy, fairness, and privacy of ML systems, assess the related engineering techniques, and evaluate their effectiveness in improving one quality target while maintaining the others. The study employs several classification models and benchmark datasets covering different contexts.

Contents

Abstract **ii**

1 Introduction **5**

2 Accuracy, fairness, privacy in ML systems **7**

2.1 Machine Learning systems pipeline 7

2.2 Accuracy, fairness and privacy 9

2.3 Thesis objective 11

2.4 Related work 12

3 Evaluating accuracy, fairness and privacy **14**

3.1 Accuracy 14

3.1.1 Assessment tools 16

3.1.2 Improvement techniques 17

3.2 Fairness 19

3.2.1 Metrics 19

3.2.2 Assessment tools 21

3.2.3 Bias mitigation techniques 22

3.3 Privacy 27

3.3.1 Metrics 27

3.3.2 Assessment tools 29
3.3.3 Privacy risk mitigation techniques 30
3.4 Summary of techniques assessment 32

4 Study methodology **34**
4.1 Environment settings 34
4.2 Project structure 36
4.2.1 Datasets 36
4.2.2 Models 37
4.2.3 Metrics 38
4.2.4 Techniques 39
4.3 Design of Experiment 45

5 Experimental evaluation **47**
5.1 Baseline overview 47
5.2 Results analysis 50
5.2.1 Effect of all techniques on individual response variables 50
5.2.2 Effect of techniques on responses 72
5.2.3 Cross-Effect of techniques on responses 78
5.2.4 Effect of all techniques on all responses 90

6 Insights and limitations **95**
6.1 Results overview 95
6.2 Threats to validity 97

7 Conclusions **98**

List of Figures

2.1 Machine Learning systems engineering pipeline 8

3.1 Application of techniques in the ML system pipeline . 33

5.1 Accuracy baseline results 48

5.2 Fairness baseline results 48

5.3 Privacy baseline results 49

5.4 Balanced accuracy results 52

5.5 Precision results 54

5.6 Recall results 56

5.7 F1 score results 58

5.8 SPD results 60

5.9 EOD results 62

5.10 AOD results 64

5.11 PDTP results 66

5.12 SHAPr results 68

5.13 Differential privacy loss results 70

5.14 Accuracy techniques on accuracy 73

5.15 Fairness techniques on fairness 75
5.16 Privacy techniques on privacy 77
5.17 Accuracy techniques on fairness 79
5.18 Accuracy techniques on privacy 81
5.19 Fairness techniques on accuracy 83
5.20 Fairness techniques on privacy 85
5.21 Privacy techniques on accuracy 87
5.22 Privacy techniques on fairness 89
5.23 Correlation matrix 92
5.24 Canonical Correlation Analysis 93

List of Tables

3.1 Bias mitigation techniques in different stages of the Machine Learning system pipeline 26

4.1 Datasets for experiments 36
4.2 Datasets features 37
4.3 Metrics 38
4.4 Random Forest grid options 40
4.5 Logistic Regression grid options 40
4.6 Random Forest best hyper-parameters 41
4.7 Logistic Regression best hyper-parameter 41

5.1 Results of the Dunn test for balanced accuracy 53
5.2 Results of the Dunn test for precision 55
5.3 Results of the Dunn test for recall 57
5.4 Results of the Dunn test for f1 score 59
5.5 Results of the Dunn test for SPD 61
5.6 Results of the Dunn test for EOD 63
5.7 Results of the Dunn test for AOD 65

5.8 Results of the Dunn test for PDTP 67

5.9 Results of the Dunn test for SHAPr 69

5.10 Results of the Dunn test for differential privacy loss . . 71

5.11 Canonical correlation coefficients 91

6.1 Techniques with greatest effect on... 96

Chapter 1

Introduction

Machine Learning (ML) is a branch of Artificial Intelligence (AI), which consists of the development of algorithms that rely on large amounts of data without being specifically programmed for a task. A Machine Learning system makes predictions based on a mathematical model and finds relationships between data.

Machine Learning systems are nowadays widely spread in many application domains, including healthcare, law, economics, and finance. As they manage sensitive data, based upon which they draw automatic decisions that impact human lives, it is not sufficient to engineer them up to satisfactory levels of accuracy. There are also ethical aspects that cannot be overlooked. Fairness and privacy are today two essential quality attributes of Machine Learning systems, that need to be assessed in relation to each other.

This thesis presents an experimental study on accuracy, fairness,

and privacy of ML systems. The study assesses the engineering techniques related to these three quality attributes, and evaluates their effectiveness in improving one quality target while maintaining the others. The study employs several classification models and benchmark datasets covering different contexts.

The thesis is structured as follows.

Chapter 2 describes the relevance of the joint evaluation of accuracy, fairness, and privacy in Machine Learning systems, sets the study objectives, and surveys related work.

Chapter 3 describes techniques, metrics and tools for assessing and improving accuracy, fairness, and privacy.

Chapter 4 presents the methodology for the experimental study.

Chapter 5 presents the experiments and discusses the results and the threats to their validity.

Chapter 6 summarizes the findings of the experimental study and discusses threats to its validity.

Chapter 7 provides concluding remarks of the experimental study.

Chapter 2

Accuracy, fairness, privacy in ML systems

2.1 Machine Learning systems pipeline

Figure 2.1 shows the pipeline of activities in a Machine Learning system and the stages that make it up:

1. **Data collection and preparation:** Data from different sources are collected. These must undergo a preparation phase that consists of handling missing, duplicate and noisy values.

2. **Data partitioning:** A dataset is typically divided into three distinct sets: a *training set* for model training, a *validation set* for hyper-parameters optimization and performance evaluation during training, and a *test set* for evaluating the model's final

performance on previously unseen data.

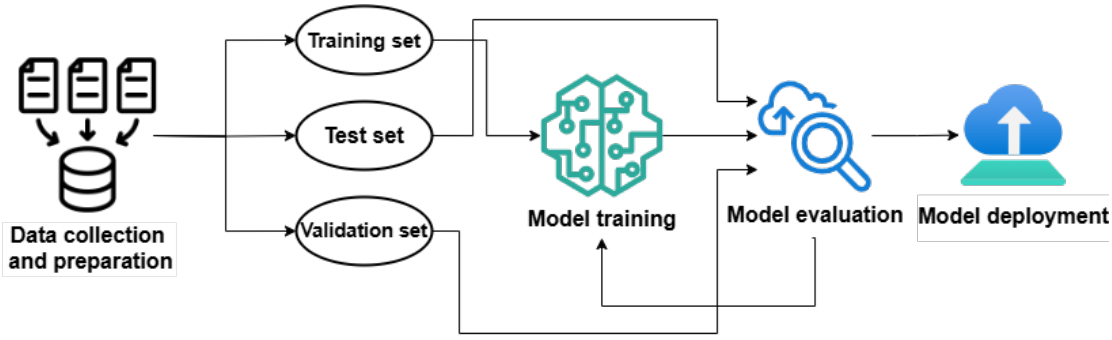

Figure 2.1: Machine Learning systems engineering pipeline

3. **Model training:** A model is trained using the training set and its performance is continuously monitored in the validation set. At this point, the most appropriate Machine Learning algorithm for the specific context can be selected.

   In the case of labeled datasets, it is referred to as **supervised learning**, i.e. the model should understand the relationships between input and output variables to make more accurate predictions about unknown data. These are used in classification and regression tasks. The most common algorithms are Logistic Regression, Linear Regression, and Neural Networks.

   Unsupervised learning occurs when the data has no labels and the model has to recognise patterns in the data by itself. It is usually used in clustering or dimensionality reduction problems.

4. **Model evaluation:** The model is evaluated in a test set, which is used to assess its performance.

5. **Model deployment:** The model is deployed to production. Then it can be integrated into an application or an automated decision-making system. Even after deployment, it is crucial that the performance of the model is monitored over time.

The described pipeline can be further enriched with pre-processing, in-processing and post-processing steps.

Pre-processing involves manipulation of the data before splitting the dataset. *In-processing* modifies the training of the model to align it with the target. *Post-processing* corrects the predictions after training the model to achieve a specific target.

2.2 Accuracy, fairness and privacy

Machine Learning systems are currently being used in various contexts, including job recruitment, credit scoring, and criminal justice.

Machine Learning models are typically trained on data that may be affected by bias [1]. This has raised concerns about the **fairness** of Machine Learning systems about ethical and legal aspects.

For example, gender bias has been demonstrated in the context of job recruitment [2]. Instead of considering the appropriateness of potential employees based on their qualifications or work experience, discriminatory practices based on gender are prevalent.

Similarly, within the context of criminal trials, which are legally required to be conducted fairly, it has been shown that the risk of

recidivism is influenced more by the race of the offender than by the nature of their previous crimes [3].

Credit scoring errors have been demonstrated for vulnerable groups defined by *protected* attributes, such as race or sex [4].

Data collected and utilized in Machine Learning systems have highlighted the need to safeguard the **privacy** of individuals, particularly when this data concerns *sensitive* information.[1] Organizations have to prevent breaches of the data they use and to ensure compliance with current regulations, such as the General Data Protection Regulation (GDPR).[2]

There are mitigation techniques in the literature to deal with data bias and privacy risks. However, these may influence the accuracy of the Machine Learning systems and/or affect each other. Not guarantee that mitigating bias does not affect privacy risks and *vice versa* that mitigating risks do not introduce bias in data, and even when mitigation means for fairness or privacy does not hurt the other aspect, nevertheless it may affect the accuracy of predictions of the Machine Learning system. In addition, accuracy improvement techniques might compromise fairness and privacy.

Thus, today a still open problem is to achieve a balance among bias mitigation, protection of individual data, and accuracy of Machine Learning systems. Although studies have examined the relation-

[1] We use the term *protected* for fairness and the term *sensitive* for privacy.
[2] https://commission.europa.eu/law/law-topic/data-protection/data-protection-eu_en

ship between accuracy and fairness or fairness and privacy, the debate persists about how these metrics should be considered collectively.

2.3 Thesis objective

The necessity to ensure that Machine Learning systems do not perpetuate inequalities or compromise privacy has resulted in the requirement of a comprehensive assessment of accuracy, fairness, and privacy.

The objective of this thesis is to provide an in-depth understanding of the effect of accuracy improvement, bias mitigation, and privacy risk mitigation techniques on the trade-off between model performance, fairness, and privacy.

Therefore, a baseline for quality targets is established first. Subsequently, each technique is applied independently to assess its effect on these metrics.

To guide this investigation, the following research questions are posed:

1. How do different mitigation and improvement techniques affect the quality targets of Machine Learning systems when applied independently?

2. What is the joint effect between the application of a technique and several response variables?

2.4 Related work

Dwork *et al.* pioneered the concept of fairness [5]. Since 2012, there has been a remarkable proliferation of literature on fairness in Machine Learning systems. A significant contribution has been made by Harman *et al.* [6]. Their survey provides a comprehensive overview of the current state of the art. It also catalogues some open source tools currently available for fairness assessment.

A further area of research concerns the effect of bias mitigation techniques and their effectiveness in improving model fairness. These works [7, 8, 9, 10] examined different techniques, highlighting a crucial issue regarding attribute analysis. The question concerns the fairness evaluation by considering a single protected attribute at a time or several protected attributes simultaneously. This choice may influence the analysis of the results. This is especially critical in cases where there are intersections between vulnerable groups.

The causal reasoning model has also been applied to identify and rectify any bias in cause-and-effect relationships between two attributes [11, 12, 13, 14].

These studies demonstrate that fairness is a multifaceted concept that should use sophisticated methodologies to ensure a comprehensive and effective examination. Despite the availability of tools and techniques, it has been demonstrated that as accuracy improves, so does unfairness, and *vice versa* [1, 15, 16, 17]. There is still considerable un-

certainty as to what the optimal method is for balancing fairness and accuracy, especially when multiple protected attributes are involved.

The concept of differential privacy is currently considered the gold standard for formally guaranteeing levels of privacy in an algorithm.

The work of Dwork *et al.* [18] is currently adopted as a point of convergence for researchers in the field. The fundamental premise is to derive valuable insights from a population while maintaining the confidentiality of individual data.

The choice between a centralized [19] and a local [20, 21, 22] approach to differential privacy depends on whether or not a trusted server is employed. This distinction is crucial because it concerns the obfuscation of user data on the client side before its transmission to the server. Recently, the impact of causal discovery on differential privacy has been studied[12].

The relationship between differential privacy and other quality metrics remains unchallenged.

Chapter 3

Evaluating accuracy, fairness and privacy

3.1 Accuracy

A model's accuracy shows how well it can classify or predict information. Accuracy is often the primary goal of an automated decision system, although it must be evaluated with other quality objectives, such as privacy and fairness. Improvements in accuracy can occasionally come at the price of these other objectives. The high accuracy of a model does not ensure its fairness or privacy protection. Let FP represent False Positives, FN represent False Negatives, TP represent True Positives, and TN represent True Negatives. The subsequent measures are defined:

- **Balanced Accuracy:** It is the mean of the true negatives and

true positives.

$$BalancedAccuracy = \frac{\left(\frac{TP}{(TP+FN)} + \frac{TN}{(TN+FP)}\right)}{2} \tag{3.1}$$

Calculating its value is simple, but it can be challenging, especially in datasets that are disproportional, meaning that one class dominates the others. Its optimal value is 1.

- **Precision:** This metric measures the ratio of the predictions recognized as correct to all predictions identified as true.

$$Precision = \frac{TP}{(TP + FP)} \tag{3.2}$$

Its ideal value is 1.

- **Recall:** This metric calculates the proportion of forecasts that were found to be true to those that were found to be accurate.

$$Recall = \frac{TP}{(TP + FN)} \tag{3.3}$$

Its ideal value is 1.

- **F1-Score**: This metric is evaluated as the harmonic mean of precision and recall.

$$F1score = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \tag{3.4}$$

It is especially useful when the classes in the dataset are not balanced because it ensures that neither precision nor recall are overemphasized. The first is its ideal value. Its optimal value is 1.

3.1.1 Assessment tools

To evaluate the performance of Machine Learning systems, several libraries have been developed. They have distinct features according to the tasks for which they were designed. The most popular ones are:

- **Scikit-learn [23]:** It is a Python library offering tools for data mining and analysis. It is based on Matplotlib [24], SciPy [25], and NumPy [26]. It offers supervised and unsupervised learning methods, such as dimensionality reduction, regression, and classification. Additionally, it enables the use of model assessment methods like Cross Validation and metrics to evaluate model performance.

- **TensorFlow [27]:** This open-source framework for training deep learning models was developed by Google. It uses pre-trained models, pre-built datasets, and GPUs to enhance computational performance.

- **PyTorch [28]:** In both academia and industry, this is another popular open source framework for neural network activities.

Like TensorFow, it supports GPUs and offers dynamic computational networks for faster model creation.

- **Keras [29]:** This intuitive API was created for those who are not familiar with neural networks. It interacts with TensorFlow features and supports both CPUs and GPUs.

3.1.2 Improvement techniques

For Machine Learning systems, achieving high accuracy is a challenging objective. As a result, specific techniques can enhance model performance. However it's crucial to understand how they affect the other quality goals. The following are the most popular methods for increasing accuracy:

- **Cross Validation [30]:** It ensures a more robust evaluation of a model's performance by reducing the risk of *overfitting*, i.e. the model fitting too closely to the training data. This technique consists of dividing the dataset into several *folds* and rotating the training set and test set among them. For instance, in a **k-fold Cross Validation** the dataset is divided into k folds and for k iterations, k-1 folds are used for training while the remaining one is for testing. The results will be averaged to obtain a more robust estimate of model performance. In this way, Cross Validation is also a way to compare various models and choose the best one.

- **Feature Engineering [31]:** Redundancy or an excessive number of features might affect a system's accuracy. By removing unnecessary elements, this approach maintains the elements that most accurately reflect the system. This increases model accuracy, decreases the probability of overfitting, and speeds up training. One possible method is Principal Component Analysis (PCA), a state transformation that enables to identification of how many components are required to represent a dataset sufficiently according to the variance explained. If fewer components are chosen than the starting dataset, PCA allows for a size reduction.

- **Hyper-parameters Tuning [32]:** During Cross Validation, it is possible to test model performance by modifying hyper-parameters value to choose the best-performing ones. One of the possible ways is to define a space of hyper-parameters and explore it according to various criteria, e.g. randomly (Randomized Search) or by testing all combinations (Grid Search).

3.2 Fairness

The equitable treatment of individuals and groups by a Machine Learning system is referred to as fairness [6].

In light of the evidence indicating the existence of bias about protected attributes such as race, age, and gender [1, 10], careful assessment and mitigation of fairness-related risks must be conducted to ensure that ML systems do not unintentionally reinforce this bias.

3.2.1 Metrics

Several indicators have been created to measure fairness. Each of these focuses on a different aspect of this quality goal.

These metrics can be generally classified into two categories: **individual fairness**, which emphasizes treating similar individuals similarly, and **group fairness**, which aims to ensure equitable outcomes across different demographic groupings.

Individual fairness metrics include:

- **Counterfactual Fairness:** It determines if a change in an individual's demographics would not impact that individual's outcome.
- **Fairness Through Awareness:** It entails establishing a fairness criterion that takes into account how similar people are to one another.

- **Fairness Through Unawareness:** It states that decisions made without considering protected attributes could be viewed as equitable.

The principal group fairness metrics are[1]:

- **Statistical Parity Difference (SPD):** It measures the difference between privileged and unprivileged groups' chances of achieving a favorable result. Mathematically, SPD is defined as follows:

$$SPD = P[\hat{Y} = 1|PA = 0] - P[\hat{Y} = 1|PA = 1], \tag{3.5}$$

  where PA stands for protected attribute, $\hat{Y}$ for the favorable prediction. The value 0 indicates total fairness.

- **Equal Opportunity Difference (EOD):** It assesses the extent to which all demographic groups exhibit comparable True Positive Rates (TPR). The EOD is calculated as follows:

$$EOD = TPR(PA = 0) - TPR(PA = 1), \tag{3.6}$$

  where $TPR = \frac{TP}{(TP+FN)}$.

- **Average Odds Difference (AOD):** It is calculated as the average of the differences between True Positive Rates (TPR) and False Positive Rates (FPR) for unprivileged and privileged

groups. Its ideal value is 0.

The AOD is computed as:

$$AOD = 0.5 * [(TPR(PA = 0) - TPR(PA = 1)) \tag{3.7}$$

$$+(FPR(PA = 0) - FPR(PA = 1)),] \tag{3.8}$$

where $FPR = \frac{FP}{(FP+TN)}$.

3.2.2 Assessment tools

Several tools exist to assess and mitigate bias in Machine Learning models. A selection of these is provided below:

- **AI Fairness 360 (AIF360)**[1]: Developed by IBM, AIF360 is an open source toolkit that provides a comprehensive suite of metrics to test bias models and datasets. It is designed for practitioners who wish to ensure fairness in their Machine Learning systems. It integrates pre-, in-, and post-processing techniques to mitigate bias.

- **Themis** [33]: It is an automated technique designed to detect discrimination in Machine Learning models. It allows users to specify fairness criteria and then evaluates the model's compliance with them by generating synthetic data.

[1]https://aif360.res.ibm.com/

- **Aequitas** [34]: It is an open-source tool for creating bias reports. Its goal is to highlight any possible difference in model outcomes between protected attributes.

- **Fairway** [15]: It enables users to apply criteria for fairness while a Machine Learning model is being trained. This guarantees that the final model maintains predictive accuracy while meeting predetermined fairness criteria.

3.2.3 Bias mitigation techniques

To mitigate potential bias in data, a number of techniques have been designed [9]. Each of these can be used at a certain point in a machine learning system's pipeline. Each has unique characteristics that enable researchers to adapt their methodology to meet the specific needs of each application.

Table 3.1 summarizes the techniques currently available and the stage in which they can be applied.

Pre-processing techniques

Pre-processing techniques are applied before to the dataset's splitting. These methods aim to reduce the bias that may influence model outcomes by changing the input data [35]:

- **Reweighing:** The learning method assigns a unique weight to each group in the dataset to ensure that each group is treated

equally without affecting the actual data points.

- **Disparate Impact Remover:** Modifies the value of features to ensure fairness while preserving the maximum possible utility of data.

- **Learning Fair Representations (LFR):** Transforms the original features into a latent representation that obscures protected attributes while retaining information relevant to the prediction task.

- **Optimized Preprocessing (OptimPreproc):** Generates an optimized dataset to reduce bias. This method considers both fairness and accuracy during data transformation, producing a dataset that is more balanced and less likely to lead to biased outcomes.

In-processing techniques

These techniques are applied during the model training phase to influence the learning algorithm to be fair.

- **Prejudice Remover:** It employs regularization by adding a penalty to the learning process to minimize bias in the model while maintaining its ability to predict.

- **Adversarial Debiasing:** It involves training a model to predict the outcome and simultaneously training an adversary to predict

protected attributes. The model learns to produce predictions less dependent on protected attributes, thus reducing bias.

- **ART Classifier:** From the Adversarial Robustness Toolbox [36], it integrates fairness constraints into the model's objective function. It helps in mitigating bias by incorporating adversarial techniques during the training process.

- **Meta Fair Classifier:** It employs a fairness constraint based on the meta-learning approach. This classifier is designed to generalize across different fairness criteria, allowing flexibility in achieving fair outcomes.

- **Gerry Fair Classifier:** It focuses on treating fairness as an optimization problem, utilizing game theory to ensure that no group is disproportionately disadvantaged by the model's predictions.

- **Exponentiated Gradient Reduction:** It reduces bias by iteratively adjusting the model's weights. This method seeks to achieve an optimal trade-off between fairness and accuracy.

- **Grid Search Reduction:** It applies a grid search to a set of models with varying degrees of fairness constraints. Then, it selects the model that provides the best balance between fairness and accuracy.

Post-processing techniques

To ensure fairness, these techniques are used to adjust the model predictions after it has been trained, avoiding the need to retrain the model.

- **Calibrated Equalized Odds Postprocessing:** It adjusts the decision thresholds of the model to equalize the odds between different groups while maintaining calibration.
- **Equalized Odds Postprocessing:**It alters the results to ensure that groups that have different protected attributes have equal rates of true positives and false positives.
- **Reject Option Classification:** It modifies the model's predictions according to a "reject option". Predictions that fall within a certain confidence interval are adjusted to ensure that the model treats each group equally.
- **Deterministic Reranking:** It reorders the predicted outcomes to ensure fairness. This method re-ranks the predictions based on fairness constraints while preserving the model's original ranking as much as possible.

Table 3.1: Bias mitigation techniques in different stages of the Machine Learning system pipeline

Stage	**Technique**
Pre-processing	Reweighing
Pre-processing	Disparate Impact Remover
Pre-processing	Learning Fair Representations
Pre-processing	Optimized Preprocessing
In-processing	Prejudice Remover
In-processing	Adversarial Debiasing
In-processing	ART Classifier
In-processing	Meta Fair Classifier
In-processing	Gerry Fair Classifier
In-processing	Exponentiated Gradient Reduction
In-processing	Grid Search Reduction
Post-processing	CalibratedEqOddsPostprocessing
Post-processing	EqOddsPostProcessing
Post-processing	Reject Option Classification
Post-processing	Deterministic Reranking

3.3 Privacy

Privacy protection must be incorporated into the development of Machine Learning systems. It is crucial to assess privacy issues and implement mitigation techniques to ensure that personal information is protected.

3.3.1 Metrics

Privacy metrics include:

- **Differential Privacy Loss [37]:** It provides a formal guarantee that individual data points cannot be readily identified. It measures the impact of adding or removing a single data point on a model's output.

  A random algorithm A satisfies (ε)-Differential Privacy if, for any two datasets D and D' that differ in exactly one element, and for any subset of outputs $S \subseteq Range(A)$:

  $$\Pr[A(D) \in S] \leq e^{\varepsilon} \Pr[A(D') \in S], \tag{3.9}$$

  where ε represents the privacy loss parameter. Smaller values of ε indicate stronger privacy guarantees. For strict privacy, ε should be close to 0, often in the range of 0 to 1.

- **SHAPr Membership Risk Privacy [38]:** It is a privacy metric based on the Shapley value, which has its roots in game theory. It quantifies each player's contribution to the coalition's total award.

  This metric assesses the risk that SHAP values could reveal whether a specific individual's data was used in the training set. Formally, the risk R_{SHAP} is the maximum difference in probabilities that a data point was or wasn't part of the training set, given its SHAP values:

$$R_{\text{SHAP}} = \max_{x_i} |\Pr(x_i \in D \mid \phi(x_i)) - \Pr(x_i \notin D \mid \phi(x_i))| \tag{3.10}$$

  The optimal value for R_{SHAP} is near zero, which indicates that there is little chance that SHAP values can be utilized to determine a person's membership status inside the training set.

- **Pointwise Differential Training Privacy (PDTP) [39]:** During the model training process, this privacy measure aims to guarantee the preservation of individual data within the training set. The PDTP measures the privacy loss related to adding or removing a particular data point from the training set. The primary goal is to provide privacy assurances for individual data entries. It aims to ensure that the presence or absence of any single data point in the training set does not significantly alter

the output of the trained model. The following is the formal definition of PDTP:

$$\epsilon_{\text{PDTP}} = \max_{x_i \in D} \left| \log \frac{\Pr(\mathcal{M}(D) = o \mid x_i \in D)}{\Pr(\mathcal{M}(D) = o \mid x_i \notin D)} \right|, \tag{3.11}$$

where $\mathcal{M}(D)$ denotes the model trained on dataset D, and o represents the output of the model. The parameter ϵ_{PDTP} is the privacy budget, and a smaller value indicates stronger privacy protection. Ideally, the value of ϵ_{PDTP} should be as low as possible, typically within the range of $[0.01, 1]$, depending on the privacy requirements and the specific application context.

3.3.2 Assessment tools

To assist practitioners in assessing and mitigating privacy risks, several specialized tools have been developed:

- **AI Privacy 360[2]:** An open-source toolkit from IBM, it provides a suite of algorithms designed to help developers understand and enhance the privacy of their models. The toolkit includes implementations of Anonymization and Minimization packages and methods for secure model evaluation.
- **Adversarial Robustness Toolbox (ART) [36]:** Also developed by IBM, it provides a set of tools designed to defend against

[2]https://aip360.res.ibm.com/

adversarial attacks. ART prioritizes robustness, but it also incorporates privacy protections such as defenses against *Membership Inference Attacks*. These attacks pose a serious threat to privacy because they try to identify whether a particular person's feature was present in the training sample.

- **Diffprivlib [40]**: This Python library was designed by IBM for implementing Differential Privacy in tasks related to data analysis and Machine Learning. Diffprivlib offers the tools and methods that let organizations perform analyses on datasets while guaranteeing the privacy of the individuals within. Due of the library's easy compatibility with well-known Machine Learning frameworks like Scikit-learn, it is user friendly for people without extensive knowledge of Differential Privacy.

3.3.3 Privacy risk mitigation techniques

A variety of privacy techniques exist to mitigate the privacy risks:

- **Differential Privacy mechanisms [41]:** These transform datasets by adding noise to obfuscate sensitive attributes. Many mechanisms exist, and each of them has different mathematical features. Two of them are:
  - **Gaussian Mechanism:** It adds normally distributed noise to the model outputs or gradients, characterized by mean

0 and standard deviation σ:

$$A(x) = f(x) + N(0, \sigma^2) \tag{3.12}$$

σ should be large enough to ensure privacy but balanced to maintain model performance.

– **Laplace Mechanism:** Noise derived from the Laplace distribution is introduced:

$$A(x) = f(x) + Lap(\frac{\Delta f}{\varepsilon}), \tag{3.13}$$

where Δf is the sensitivity of the function f. A smaller ε and a appropriate Δf yield better privacy, with minimal impact on accuracy.

- **ML Anonymization [42]:** This method modifies the dataset before training by making sure that any data used to train the model cannot be readily linked to a specific person.

- **Privacy Risk Assessment [43]:** Understanding the privacy vulnerabilities in ML models is a prerequisite for performing attacks. It is important to prevent an attacker from being able to detect whether or not a particular item of data is part of the training set. The aim is to minimise the potential exposure of personal information to attacks. There are reliable resources for

evaluating and reducing such risks in the Adversarial Robustness Toolbox (ART). Machine Learning programs have to employ this technique to safeguard data and adhere to privacy laws.

3.4 Summary of techniques assessment

The preceding paragraphs may be summarized by Figure 3.1, which illustrates the pipeline of a Machine Learning system and embraces improvement and mitigation techniques discussed previously.

Once the data have been collected and prepared, it is possible to apply pre-processing techniques, including those related to fairness, feature engineering, and differential privacy mechanisms, to modify the initial dataset. Subsequently, Cross Validation may be employed in dataset partition to enhance the resilience of the over-fitting. Subsequently, techniques for mitigating bias and anonymization can be employed. Additionally, the model hyper-parameters can be selected to align with the system's objectives. Following training, post-processing techniques can be utilized to rectify any biases and assess the system's reaction to attacks. Ultimately, a final evaluation and deployment can follow.

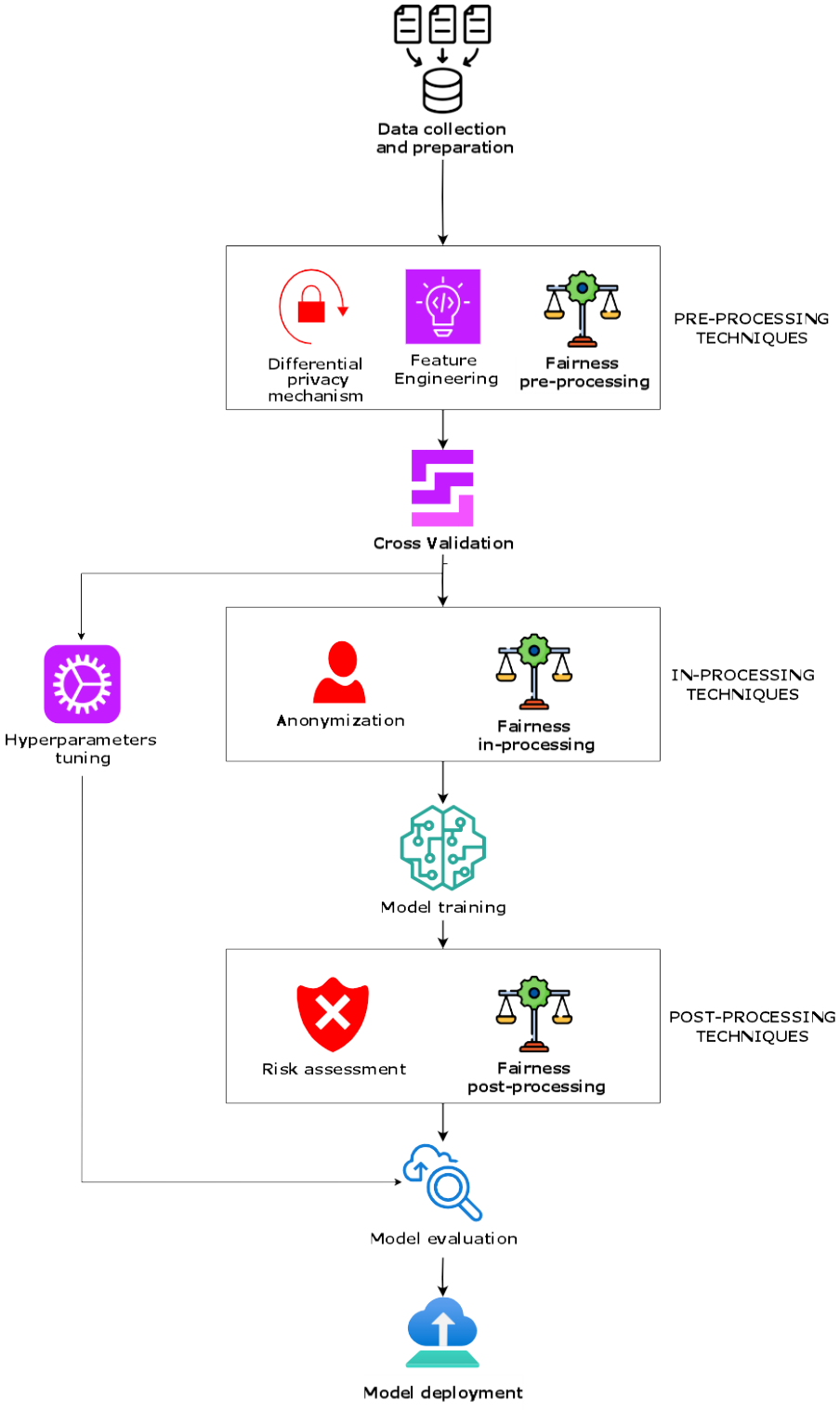

Figure 3.1: Application of techniques in the ML system pipeline

Chapter 4

Study methodology

4.1 Environment settings

The objective is to create a custom code that, by choosing a dataset, a model, and a technique, returns the metrics values of all three categories for that combination.

The hardware platform on which the code was developed is a laptop with an Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz, 8GB RAM memory and 64-bit Windows 11 home operating system.

Python was chosen as the programming language for its versatility and widespread use in the literature, as well tools and libraries analyzed. In addition, a lightweight virtual environment (*venv*) was chosen to encapsulate the packages to be installed.

The idea was to create a modular code for a clear separation of functionalities and easier handling of changes. The project is struc-

tured in the following files:

- *requirements.txt*, file containing the packages to be installed.
- *data*, folder containing raw data to be placed in the appropriate sub-folder inside the virtual environment folder.
- *dataset*, folder containing the file to load and prepare the datasets.
- *metrics*, folder containing three files to calculate the accuracy, fairness, and privacy metrics separately.
- *models*, folder containing the file for loading classification models.
- *main.py*, main program that implements the logic of the Machine Learning system pipeline.
- *techniques.py*, file containing function calls of the implemented techniques.
- *run_techniques.sh*, script to repeatedly launch different input combinations and save the results in text files.
- *results.py*, program to save all results in one excel sheet.
- *results*, folder containing text file outputs.

Among the open source tools illustrated in the previous chapter, **Scikit-learn** was chosen for the accuracy assessment and **IBM**

Trusted AI Technologies (AIF360, AI Privacy 360, Adversarial Robustness 360 Toolbox and Diffprivlib) for fairness and privacy assessment. The choice was made on these for their wide use in the literature since the resources made available by IBM were considered comprehensive in all their aspects.

4.2 Project structure

4.2.1 Datasets

Three datasets widely used in the literature were chosen to be studied: Adult, Compas, and German. These were chosen because they differ in size and cover different application domains.

Table 4.1 shows the summary information of the datasets. All of them are binary classification tasks integrated within the IBM AIF360 toolkit: in Adult the prediction of income greater or less than 50k; in Compas the risk or not of recidivism; in German the assessment of good or bad credit risk.

Table 4.1: Datasets for experiments

Dataset	**Ref**	**Source**	**Year**	**# instances**	**Content**
Adult	[44]	UCI Machine Learning Repository	1996	48,842	US census data
Compas	[45]	ProPublica	2016	7,214	Florida criminal data
German	[46]	UCI Machine Learning Repository	1994	1,000	German loan requests data

Table 4.2: Datasets features

Dataset	**Protected attribute(s)**	**Privileged group(s)**	**Favorable class**
Adult	Sex, Race	Sex=Male, Race =White	Income>50k
Compas	Sex, Race	Sex=Male, Race=Caucasian	No recidivism
German	Sex	Sex=Male	Good credit

Table 4.2 shows for each dataset protected attributes, privileged groups, and favorable class.

Datasets are subjected to a default pre-processing belonging to the AIF360 toolkit. Operations include data cleaning, handling of missing values, and preparation for fairness analysis.

4.2.2 Models

Two classification models – **Logistic Regression (LSR)** [47] and **Random Forest (RF)** [48] - were chosen to compare performance.

Logistic Regression is a simple method for binary classification. Random Forest is an ensemble learning technique that combines the results of several decision trees. Both models are suitable for handling the binary classification tasks defined for each dataset in this study. The choice is mainly due to their compatibility with the chosen tools. In addition, previous research [1, 10, 49, 50] favoured these models for their simplicity over neural networks.

Although LSR is a relatively simple method, it can have limitations when dealing with several independent variables or when classes are

highly unbalanced. The maximum number of iterations was set to 1,000 for this experimentation. In addition, the *lbfgs* solver was chosen to avoid memory problems.

For RF the default settings were chosen: 100 estimators, the *gini* criterion for the quality of the split, 2 as the minimum number of samples to split a node and 1 as the minimum number of leaf nodes.

4.2.3 Metrics

A total of ten metrics were chosen to be evaluated, as shown in Table 4.3. Specifically, the accuracy metrics are evaluated with Scikit-learn, the fairness metrics with AIF360, the PDTP and SHAPr Membership Privacy Risk metrics with ART, and the Differential Privacy Loss with diffprivlib.

In calculating the SHAPr, specific reasoning was performed according to the model used, as this metric is based on the importance of features and thus reflects not only the presence of the record in the training dataset but also how much each feature contributes to the model's predictions. It is useful to identify which aspects of the data

Table 4.3: Metrics

Accuracy	**Group fairness**	**Privacy**
Balanced accuracy	SPD	PDTP
Precision	EOD	Differential Privacy loss
Recall	AOD	SHAPr Membership Privacy Risk
F1-score		

may be more sensitive and thus exposed to privacy risks.

With Logistic Regression, the distances between predictions are no longer based on the geometric close neighborhood, but on changes in class probabilities. Probabilities are calculated for each test sample; the challenge is to evaluate how probabilities change when a training sample is removed.

With Random Forest, it is evaluated how the importance of features changes when a training sample is removed.

The average value of these differences has provided the measure of leakage.

4.2.4 Techniques

Thirteen techniques were chosen to be implemented at different stages of the ML pipeline.

The techniques chosen to **improve accuracy** are:

- **Cross Validation:** A random division into 5 folds was chosen and the random state was set at 42 to check randomness of experiments. This is an arbitrary number, chosen by programmers as an omen to Douglas Adams‘ novel [51], according to which the number 42 answers all questions about life and universe.

- **PCA:** First, the dataset was split into train and test with an 80:20 ratio and the data were scaled. Since the datasets consid-

ered have different dimensions, it was decided to set the number of components to preserve 0.9 of the variance.

- **Hyper-parameters tuning:** First, a 5-fold Cross Validation was performed. Subsequently, a Grid Search approach was chosen with Table 4.4 for RF and Table 4.5 for LSR:

Table 4.4: Random Forest grid options

Parameter	**Value 1**	**Value 2**	**Value 3**	**Value 4**
# estimators	50	100	200	
Max depth	5	10	20	None
Min samples split	2	5	10	
Min samples leaf	1	2	4	

Table 4.5: Logistic Regression grid options

Cs	**CV**	**Solver**	**Scoring**	**Random state**	**Max iter**
10	5	lbfgs	f1	42	1,000

Table 4.6 shows the best RF hyper-parameters found.

In the case of LSR, the regularization parameter Cs was set equal to 10, which means that Scikit-learn automatically selected a logarithmic range of values between 10^{-4} and 10^{4}. Table 4.7 shows the best value of C found for each dataset:

Table 4.6: Random Forest best hyper-parameters

Dataset	**# estimators**	**Max depth**	**Min samples split**	**Min samples leaf**
Adult	200	None	2	2
Compas	100	None	10	1
German	200	None	5	1

Table 4.7: Logistic Regression best hyper-parameter

Dataset	**Best C found**
Adult	21,5443469
Compas	0,04641589
German	0,35938137

Regarding **bias mitigation**, the effect of six techniques compatible with the chosen models was evaluated:

- **Reweighing (Pre-processing):** The parameters to be passed are the privileged and non-privileged groups for each dataset, specified in Table 4.2. The reweighted transformed dataset is returned. The pipeline follows.

- **Disparate Impact Remover (Pre-processing):** It modifies feature values by increasing the group fairness indicating 1.0 as full repair and a protected attribute. In the case of Adult and Compas datasets, for which two protected attributes were specified, the technique was applied to each one. The pipeline follows.

- **Grid Search Reduction (In-processing):** After loading the model, the *Demograpic Parity* fairness constraint was chosen. Next, the classifier object is prepared with the estimator, i.e. the fitted and predicted chosen model, and the protected attributes. Other parameters are a constraint weight of 0.5 as the relative weight put on the constraint violation when selecting the best model, 10 as the grid size, and 2.0 as the largest Lagrange multiplier to be considered. The pipeline follows.

- **Exponentiated Grid Search (In-processing):** It reduces fair classification to a sequence of cost-sensitive classification problems, returning a randomized classifier with the lowest empirical error subject to fair classification constraints. After loading the model, the object is prepared with the estimator, the fairness constraint *Demographic Parity*, a 0.01 epsilon value indicating the allowed violation of the constraint, 50 iterations. The pipeline follows.

- **Calibrated EqOdds Post-processing (Post-processing):** After the 60:20:20 train-test-validation split and the model estimation, the technique attempts to correct the predictions according to the privileged and non-privileged groups in each case, generalized false negative rate (*fnr*) as a cost constraint and a fixed seed to make reproducible predictions. The estimation of the validation and the predictions on the test proceed.

- **EqOdds Post-processing (Post-processing):** Similar to the previous technique, it attempts to correct the predictions according to the privileged and non-privileged groups in each case. A seed is set to make reproducible predictions. The estimation of the validation and the predictions on the test proceed.

The techniques chosen to **mitigate privacy risk** are:

- **Laplace Mechanism:** To protect differential privacy, a Laplace noise is applied to the dataset as a pre-processing step. A privacy parameter epsilon 0.5 controls the amount of noise added. A data sensitivity of 1 defines how much a single instance can affect the output. A random state fixed at 42 ensures the reproducibility of the added noise.

- **Gaussian Mechanism:** Gaussian noise is applied to the data set during the pre-processing phase. A privacy parameter epsilon 0.5 controls the amount of added noise. The delta parameter 0.000001 indicates the tolerance for privacy violations. A data sensitivity of 1 defines how much a single instance can affect the output. A random state set at 42 ensures the reproducibility of the added noise.

- **Anonymization:** To reduce the risk of re-identification of sensitive attributes, it is set an anonymization parameter k equal to 50 and a list of *quasi-identifiers*, i.e. attributes that can identify

an individual. The list varies according to the case studied: for the Adult dataset the age, level of education, income, and weekly working hours; for Compas the case identifier; for German the duration in months of the loan, the purpose, credit history, work, and home information.

- **Membership Inference Black-Box Attack:** The objective of this attack is to detect the presence of a particular data element within the model's training set. Accordingly, the attack is prepared using both the training data and the test data. During this process, the attack learns to distinguish between the two. Once trained, the attack performs an inference on the test data to determine whether each sample was part of the model's original training set in order to assess the model's vulnerability.

If the technique is not specified (**none**), the main program evaluates the ten metrics based only on the dataset and model. The program loads the dataset, preprocesses it according to the default settings, divides it into train and test with an 80:20 ratio, and scales them. The selected model is loaded. Estimation and prediction are performed. The metrics are calculated and their value is returned.

4.3 Design of Experiment

Design of Experiment (DoE) is a systematic approach to planning, executing, and analyzing experiments [52].

An *experiment* is a test in which observing the output based on the input and acting on controllable factors, to inspect and identify the reason for the change in the output. The *response variable* is the measured output. A *factor* is a variable that influences the response. The *levels* are the number of values a factor can take. The *repeats* are the number of repetitions.

The experiment in this thesis aims to study the effects of controllable factors on the response and to understand which ones are the most influential. Therefore, in this study, the **response variables** chosen are ten and correspond to the ten metrics considered. The **factors** are:

- **DATASET:** 3 levels (*Adult, Compas, German*)
- **MODEL:** 2 levels (*LSR, RF*)
- **TECHNIQUE:** 16 levels (*None, PCA, Laplace Mechanism, Anonymization, CalibratedEqOdds, DisparateImpactRemover_SEX, EqOddsPostProcessing, CrossValidation, Gaussian Mechanism, ExponentiatedGradientReduction, DisparateImpactRemover_RACE, MembershipInferenceAttack, HyperparametersOptim, Reweighing, GridSearchReductionSEX, GridSearchReductionRACE*)

The basic idea is to consider all possible combinations of all levels of all factors. However, a separate discussion must be made: *DisparateImpactRemover_RACE* and *GridSearchReductionRACE* are only applicable to 2 levels of the factor "Dataset" instead of 3 because the dataset *German* does not have the protected attribute "race".

Finally, to improve the consistency of the results, it was decided to perform **10 replications** per combination.

Consequently, the total number of experiments conducted is:

10 repetitions$\times$3 datasets$\times$2 models$\times$14 techniques = 840 experiments

10 repetitions$\times$2 datasets$\times$2 models$\times$2 techniques = 80 experiments

Thus, **920 experiments** were conducted.

Chapter 5

Experimental evaluation

5.1 Baseline overview

If the technique to be used was not specified, the program was launched indicating only the dataset and the model to be used. In this way, it was possible to collect the results of a baseline that would later be compared with the results of the various techniques applied. It is important to note the distinction of six case studies, concerning different combinations of *Dataset-Model* pairings.

The two **vertical bar graphs** in Figure 5.1 show the values of **accuracy metrics** for the two models under analysis. Each colored bar represents a metric. The height of the bars indicates the average value for each. It can be noted that the LSR model has a more uniform performance overall for the German dataset. LSR has higher values of balanced accuracy and precision, while RF has higher recall values.

F1 score values are quite similar.

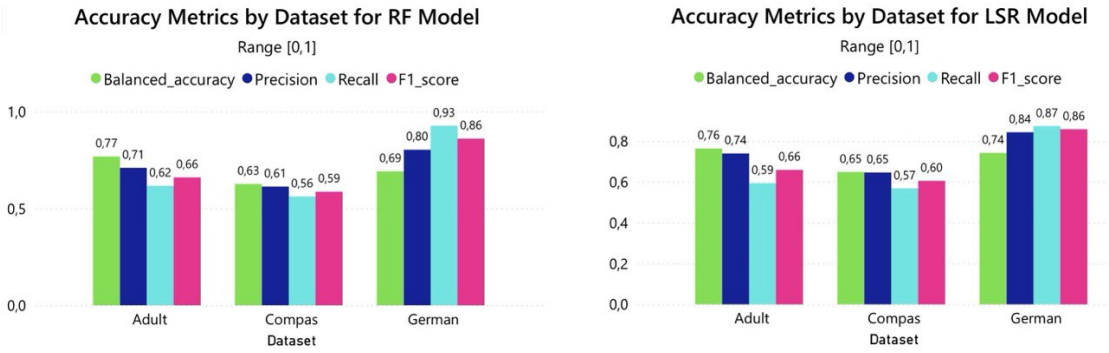

Figure 5.1: Accuracy baseline results

The **horizontal bar charts** in Figure 5.2 show the **fairness metrics** values for the two models under analysis. It should be noted that 0 indicates total fairness, while the allowable fairness range is [-0.1,0.1]. Exceeding these extremes indicates total unfairness.

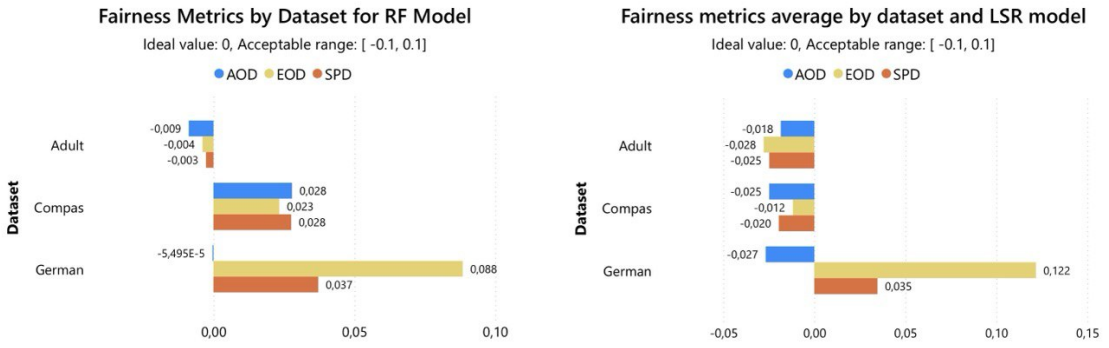

Figure 5.2: Fairness baseline results

It can be noted that for Adult both models have negative fairness values and in particular the LSR values are more unfair. For Compas the RF model resulted in positive fairness values, whereas the LSR model resulted in negative values. For German in both models EOD

and SPD are positive and quite high; in particular, it can be seen that EOD exceeds the upper limit of the permissible range of fairness. In contrast, AOD is negative in both cases but very close to 0 in the case of RF, indicating a closeness to complete fairness.

Six **area plots** in Figure 5.3 show privacy metrics. PDTP values were min-max normalized to be within the [0,1] range.

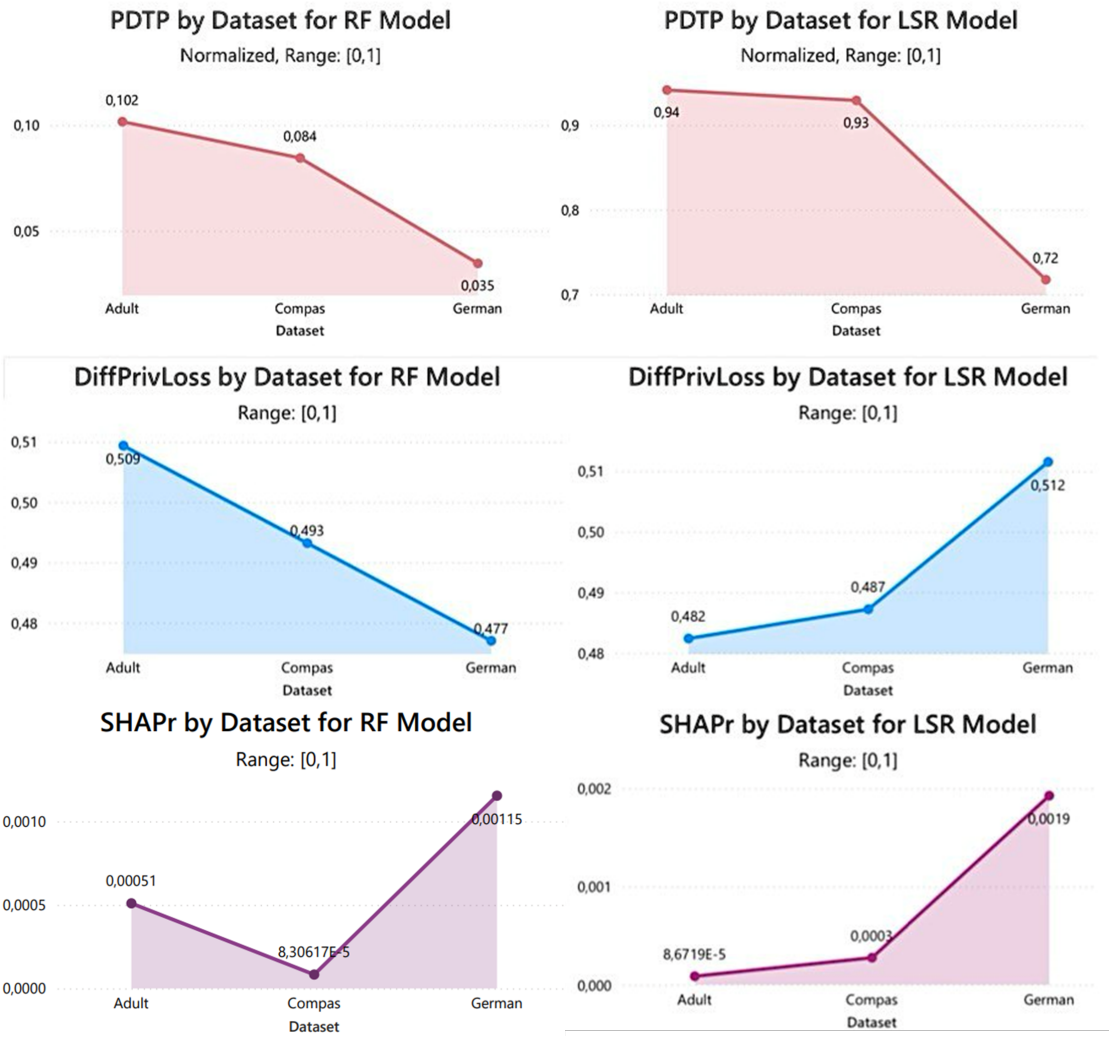

Figure 5.3: Privacy baseline results

PDTP is the highest in the case of the LSR model, while it has

lower values with RF.

DiffPrivLoss is rather low for all datasets, meaning that despite the Differential Privacy guarantee, the models lose little accuracy. It can be observed that Adult has the highest value with the RF model and the lowest with LSR, while German has the opposite behavior.

It can be noted that German has higher values for SHAPr than the other two datasets. Overall all three datasets have good values, as they are close to zero and thus indicate little privacy exposure.

5.2 Results analysis

To guide results analysis, the following **research questions** are posed:

1. How do different mitigation and improvement techniques affect the quality targets of ML models when applied independently?

2. What is the joint effect between the application of a technique and several response variables?

5.2.1 Effect of all techniques on individual response variables

For a statistical evaluation of the impact of the techniques on the individual response variables, a **Dunn test with control group** was performed, one for each combination of dataset and model.

The Dunn test is an ANOVA non-parametric statistical test that compares pairs of groups to see if there are statistical differences [53]. The control group means that no technique is applied.

The **null hypothesis** H_0 is that there are no significant differences between the control group (*None*) and each group (*Technique*). The **alternative hypothesis** H_1 is that at least one of the groups is significantly different from the control group.

The **p-value** is the smallest value for which the null hypothesis is rejected. If the p-value is less than a **significance level** α of 0.05, the null hypothesis is rejected for that group. In that case, it indicates a significant difference in the technique compared to the control group.

Below are horizontal bar graphs illustrating the impact of the techniques on the individual responses for each dataset and model pair. The blue color indicates the baseline value, the green color an improvement, the orange color a slight worsening, and the red color a marked worsening. In addition, the statistically significant p-values of the techniques compared to the control group will be reported.

Figure 5.4 and table 5.1 show the results for balanced accuracy.

It should be noted that the Compas dataset had more improvements overall than the other two. PCA had an unexpected negative impact, while Anonymization and bias mitigation techniques unexpectedly brought improvements.

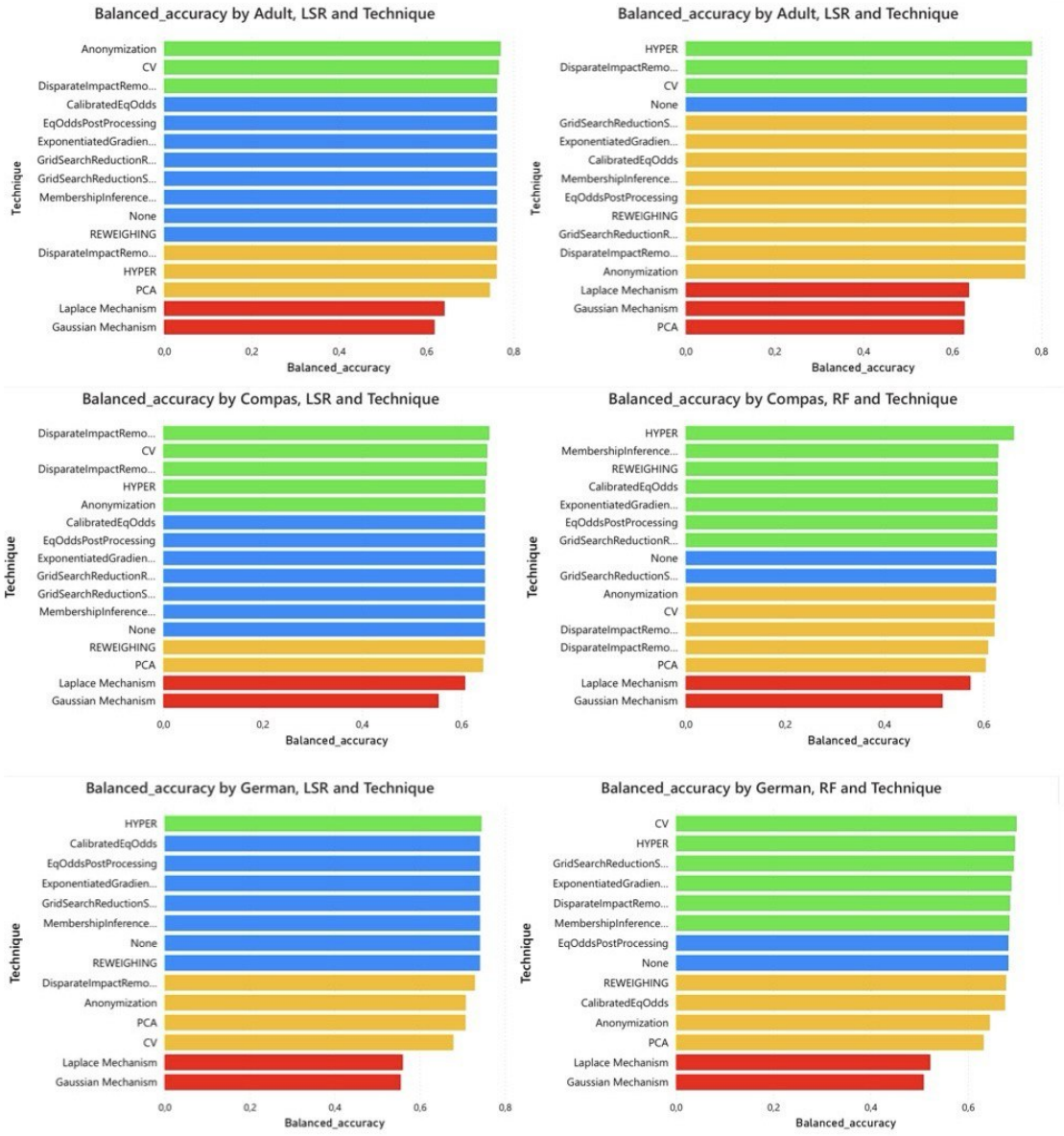

Figure 5.4: Balanced accuracy results

Table 5.1: Results of the Dunn test for balanced accuracy

Subject	**Technique**	**P-value**
Adult-LSR	Anonymization	0,0120
Adult-LSR	PCA	0,0120
Adult-LSR	Laplace Mechanism	0,0016
Adult-LSR	Gaussian Mechanism	0,0002
Adult-RF	PCA	<0,0001
Adult-RF	Laplace Mechanism	0,0006
Adult-RF	Gaussian Mechanism	<0,0001
Compas-LSR	Gaussian Mechanism	0,0120
Compas-LSR	Cross validation	0,0016
Compas-LSR	Disparate Impact Remover_Sex	0,0002
Compas-LSR	Disparate Impact Remover_Race	0,0120
Compas-RF	PCA	0,0326
Compas-RF	Laplace Mechanism	0,0050
Compas-RF	Gaussian Mechanism	0,0007
Compas-RF	HyperparamOptim	0,0235
German-LSR	Anonymization	0,0420
German-LSR	PCA	0,0053
German-LSR	Laplace Mechanism	0,0050
German-LSR	Gaussian Mechanism	0,0007
German-LSR	Cross Validation	0,0005
German-RF	Anonymization	0,0420
German-RF	PCA	0,0053
German-RF	Laplace Mechanism	<0,0001
German-RF	Gaussian Mechanism	<0,0001

Figure 5.5 and Table 5.2 show results for precision metric.

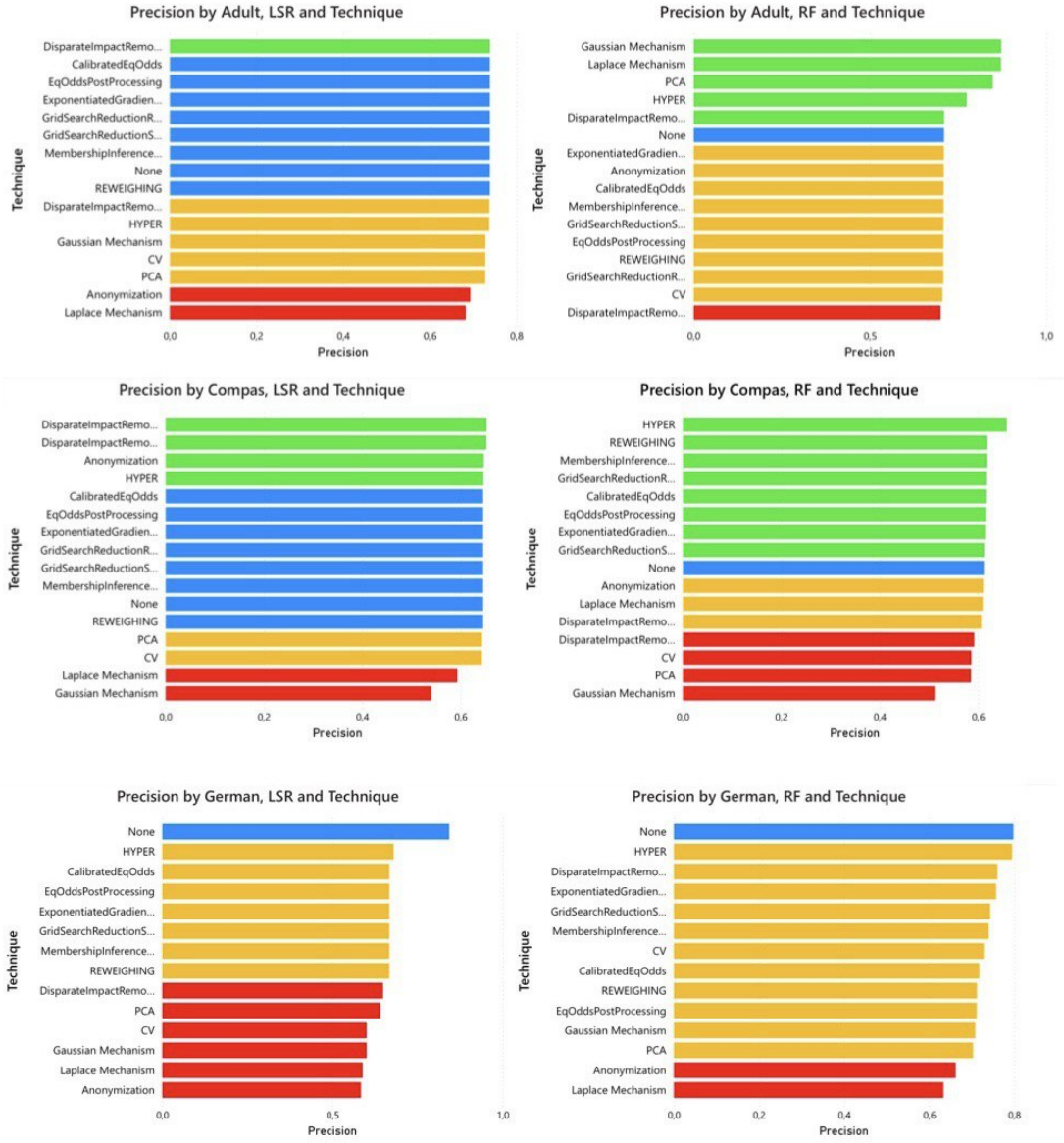

Figure 5.5: Precision results

The Compas dataset showed more overall improvements, while performance declined on the German one. Bias mitigation techniques, as Disparate Impact Remover, also brought unexpected improvements.

Table 5.2: Results of the Dunn test for precision

Subject	**Technique**	**P-value**
Adult-LSR	Anonymization	<0,0001
Adult-LSR	PCA	0,0002
Adult-LSR	Laplace Mechanism	<0,0001
Adult-LSR	Gaussian Mechanism	0,0120
Adult-LSR	Cross Validation	0,0016
Adult-RF	Disparate Impact Remover_Race	0,0119
Adult-RF	Laplace Mechanism	0,0102
Adult-RF	Gaussian Mechanism	0,0075
Compas-LSR	Gaussian Mechanism	0,0120
Compas-LSR	Laplace Mechanism	0,0016
Compas-LSR	Disparate Impact Remover_Sex	0,0016
Compas-LSR	Disparate Impact Remover_Race	0,0120
Compas-RF	PCA	0,0174
Compas-RF	Cross Validation	0,0177
Compas-RF	Gaussian Mechanism	0,0007
Compas-RF	HyperparamOptim	0,0239
German-LSR	Anonymization	<0,0001
German-LSR	PCA	<0,0001
German-LSR	Laplace Mechanism	<0,0001
German-LSR	Gaussian Mechanism	<0,0001
German-LSR	Cross Validation	<0,0001
German-LSR	Disparate Impact Remover_Sex	<0,0001
German-RF	Anonymization	<0,0001
German-RF	PCA	<0,0001
German-RF	Laplace Mechanism	<0,0001
German-RF	Gaussian Mechanism	<0,0001
German-RF	Grid Search Reduction_SEX	0,0494
German-RF	Cross Validation	0,0161
German-RF	Calibrated EqOdds	0,0081
German-RF	EqOdds PostProcessing	0,0030
German-RF	Reweighing	0,0003

Figure 5.6 and Table 5.3 show results for recall metric.

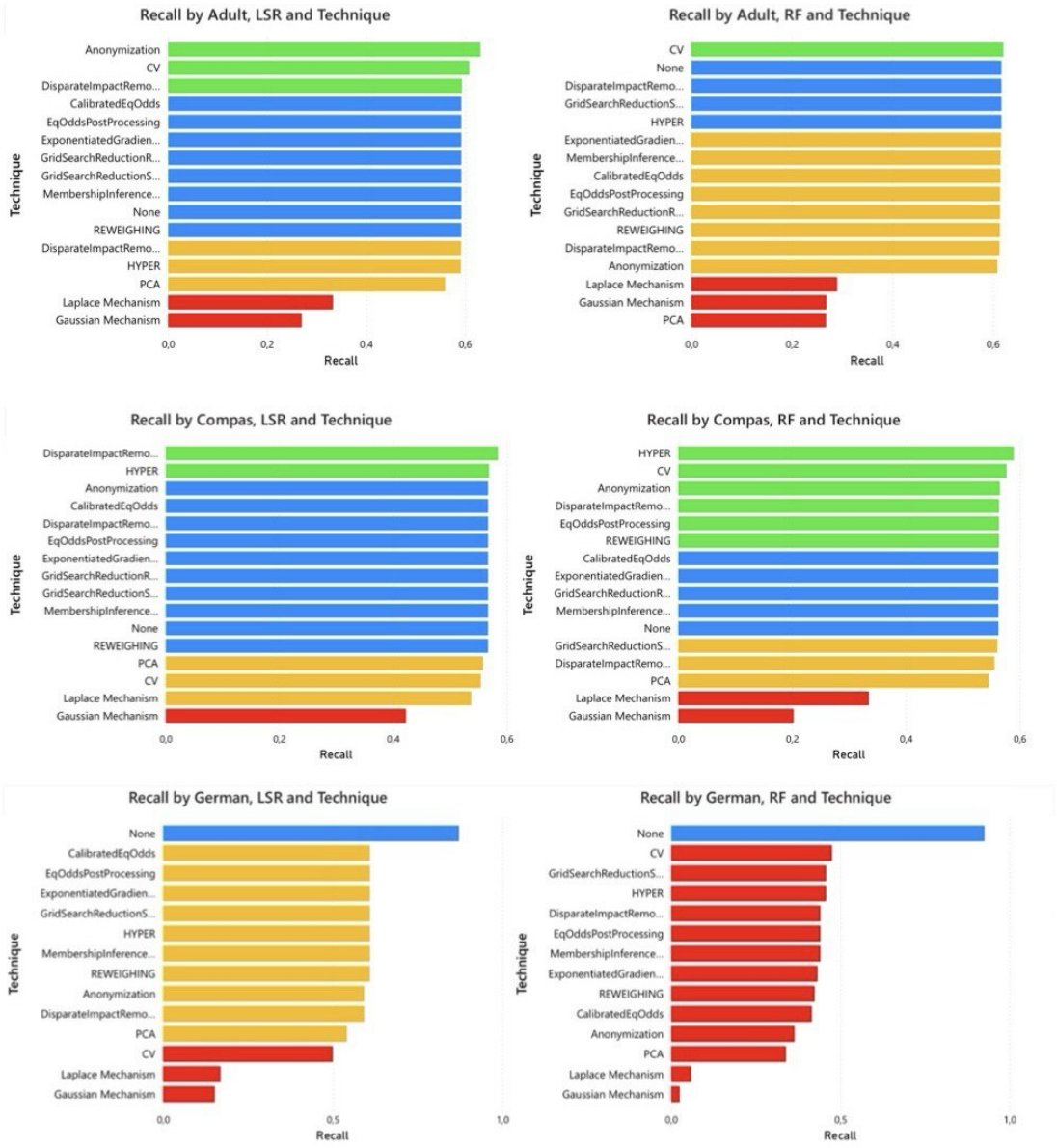

Figure 5.6: Recall results

It should be noted how the Compas dataset had more improvements overall, while German had only worsening compared to the baseline. It can be observed how unexpectedly bias mitigation and privacy mitigation techniques brought improvements.

Table 5.3: Results of the Dunn test for recall

Subject	**Technique**	**P-value**
Adult-LSR	Anonymization	0,0120
Adult-LSR	PCA	0,0120
Adult-LSR	Laplace Mechanism	0,0016
Adult-LSR	Gaussian Mechanism	0,0002
Adult-RF	Anonymization	0,0266
Adult-RF	Laplace Mechanism	0,0007
Adult-RF	Gaussian Mechanism	<0,0001
Adult-RF	PCA	<0,0001
Compas-LSR	Gaussian Mechanism	<0,0001
Compas-LSR	Laplace Mechanism	0,0005
Compas-LSR	Disparate Impact Remover_Sex	0,0047
Compas-LSR	PCA	0,0343
Compas-LSR	Cross Validation	0,0047
Compas-RF	Laplace Mechanism	0,0237
Compas-RF	Gaussian Mechanism	0,0040
Compas-RF	HyperparamOptim	0,0059
German-LSR	Anonymization	<0,0001
German-LSR	PCA	<0,0001
German-LSR	Laplace Mechanism	<0,0001
German-LSR	Gaussian Mechanism	<0,0001
German-LSR	Cross Validation	<0,0001
German-LSR	Disparate Impact Remover_Sex	<0,0001
German-RF	Anonymization	<0,0001
German-RF	PCA	<0,0001
German-RF	Laplace Mechanism	<0,0001
German-RF	Gaussian Mechanism	<0,0001
German-RF	Disparate Impact Remover_SEX	0,0036
German-RF	Exponentiated Gradient Reduction	0,0245
German-RF	Calibrated EqOdds	0,0046
German-RF	EqOdds PostProcessing	0,0243
German-RF	Reweighing	0,0046
German-RF	Membership Inference Attack	0,0360

Figure 5.7 and table 5.4 show results for F1 score.

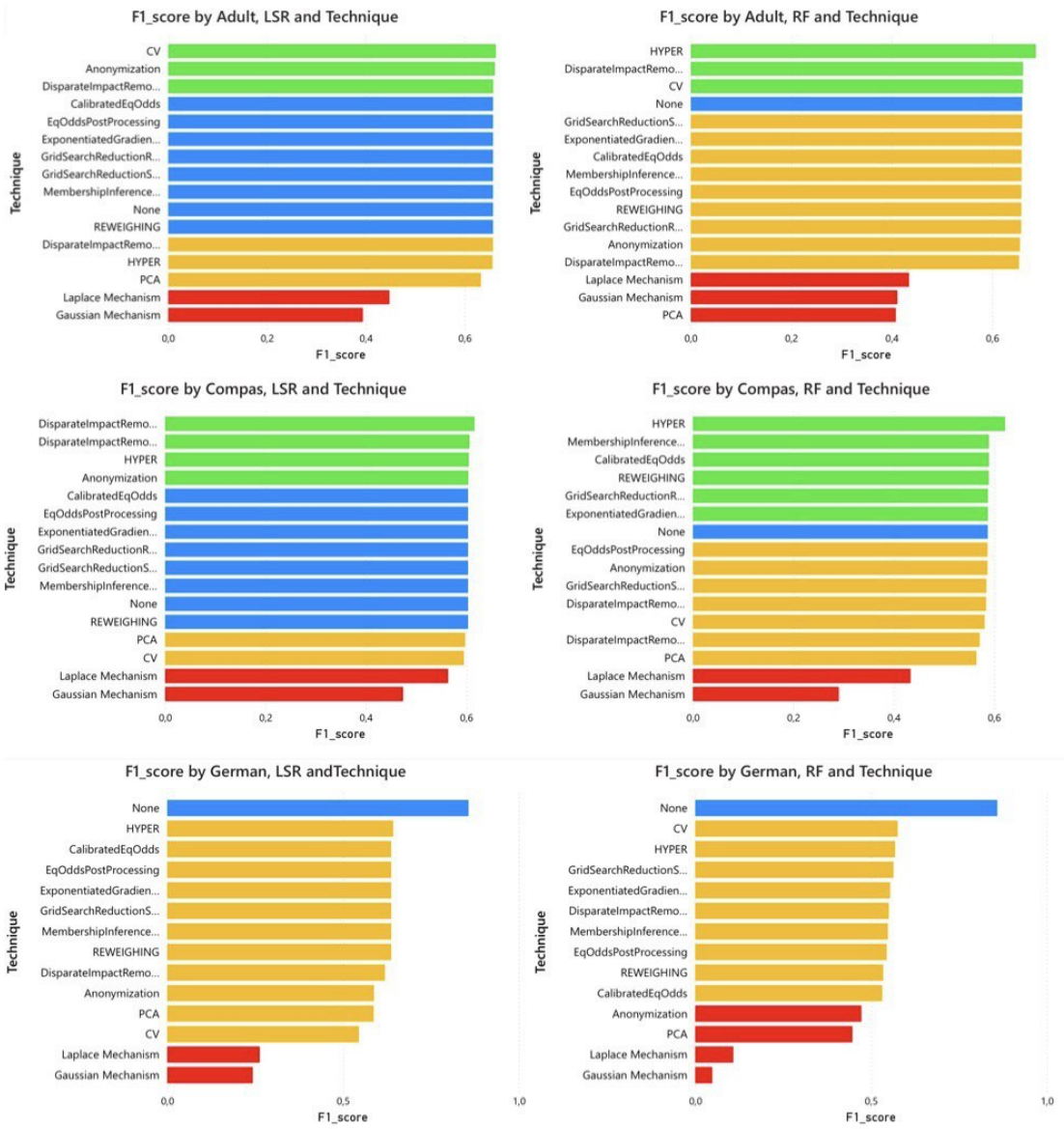

Figure 5.7: F1 score results

As a consequence of the precision and recall results, it should be noted that the Compas dataset had more improvements overall, while German had only deteriorations compared to the baseline.

Table 5.4: Results of the Dunn test for f1 score

Subject	**Technique**	**P-value**
Adult-LSR	Cross Validation	0,0120
Adult-LSR	PCA	0,0120
Adult-LSR	Laplace Mechanism	0,0016
Adult-LSR	Gaussian Mechanism	0,0002
Adult-RF	Anonymization	0,0471
Adult-RF	Laplace Mechanism	0,0007
Adult-RF	Gaussian Mechanism	<0,0001
Adult-RF	PCA	<0,0001
Compas-LSR	Gaussian Mechanism	0,0016
Compas-LSR	Laplace Mechanism	0,0120
Compas-LSR	Disparate Impact Remover_Sex	0,0016
Compas-LSR	Disparate Impact Remover_Race	0,0120
Compas-RF	Laplace Mechanism	0,0054
Compas-RF	Gaussian Mechanism	0,0008
Compas-RF	HyperparamOptim	0,0222
Compas-RF	PCA	0,0357
German-LSR	Anonymization	<0,0001
German-LSR	PCA	<0,0001
German-LSR	Laplace Mechanism	<0,0001
German-LSR	Gaussian Mechanism	<0,0001
German-LSR	Cross Validation	<0,0001
German-LSR	Disparate Impact Remover_Sex	<0,0001
German-RF	Anonymization	<0,0001
German-RF	PCA	<0,0001
German-RF	Laplace Mechanism	<0,0001
German-RF	Gaussian Mechanism	<0,0001
German-RF	Disparate Impact Remover_SEX	0,0092
German-RF	Calibrated EqOdds	0,0047
German-RF	EqOdds PostProcessing	0,0147
German-RF	Reweighing	0,0018
German-RF	Membership Inference Attack	0,0451

Figure 5.8 and table 5.5 show results for SPD metric.

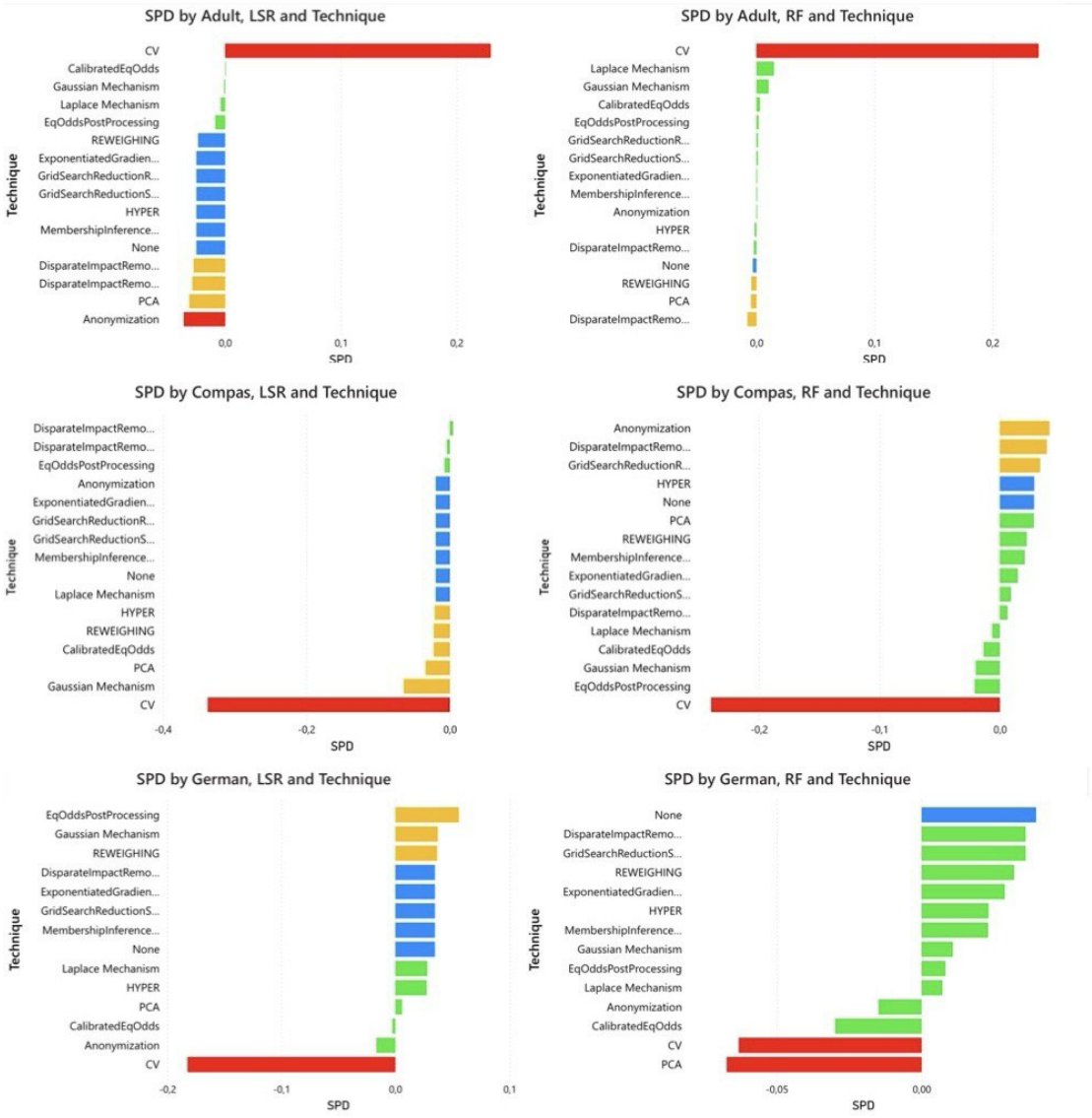

Figure 5.8: SPD results

It can be noticed that there is a general improvement in all six cases, but in particular the choice of the Random Forest as a model has also brought improvements in the case of privacy techniques and Hyper-parameters Optimization.

Table 5.5: Results of the Dunn test for SPD

Subject	**Technique**	**P-value**
Adult-LSR	Cross Validation	0,0004
Adult-LSR	Anonymization	,0191
Adult-LSR	Calibrated EqOdds	0,0283
Adult-LSR	Gaussian Mechanism	0,0161
Adult-RF	Cross Validation	<0,0001
Adult-RF	Laplace Mechanism	0,0009
Adult-RF	Gaussian Mechanism	0,0034
Compas-LSR	Gaussian Mechanism	0,0005
Compas-LSR	PCA	0,0078
Compas-LSR	Cross Validation	<0,0001
Compas-RF	Gaussian Mechanism	0,0262
Compas-RF	Cross Validation	<0,0001
Compas-RF	EqOdds Postprocessing	0,0164
German-LSR	Anonymization	0,0040
German-LSR	Cross Validation	<0,0001
German-RF	Anonymization	0,0004
German-RF	PCA	<0,0001
German-RF	Cross Validation	<0,0001

Figure 5.9 and Table 5.6 show results for EOD metric.

It can be noted that there is a general improvement in all six cases, with no particular distinction in the choice of models. It is interesting to note the positive results for German, also with reference to the accuracy and privacy techniques. In the other cases, it is mostly cross validation the only technique that has worsened the results.

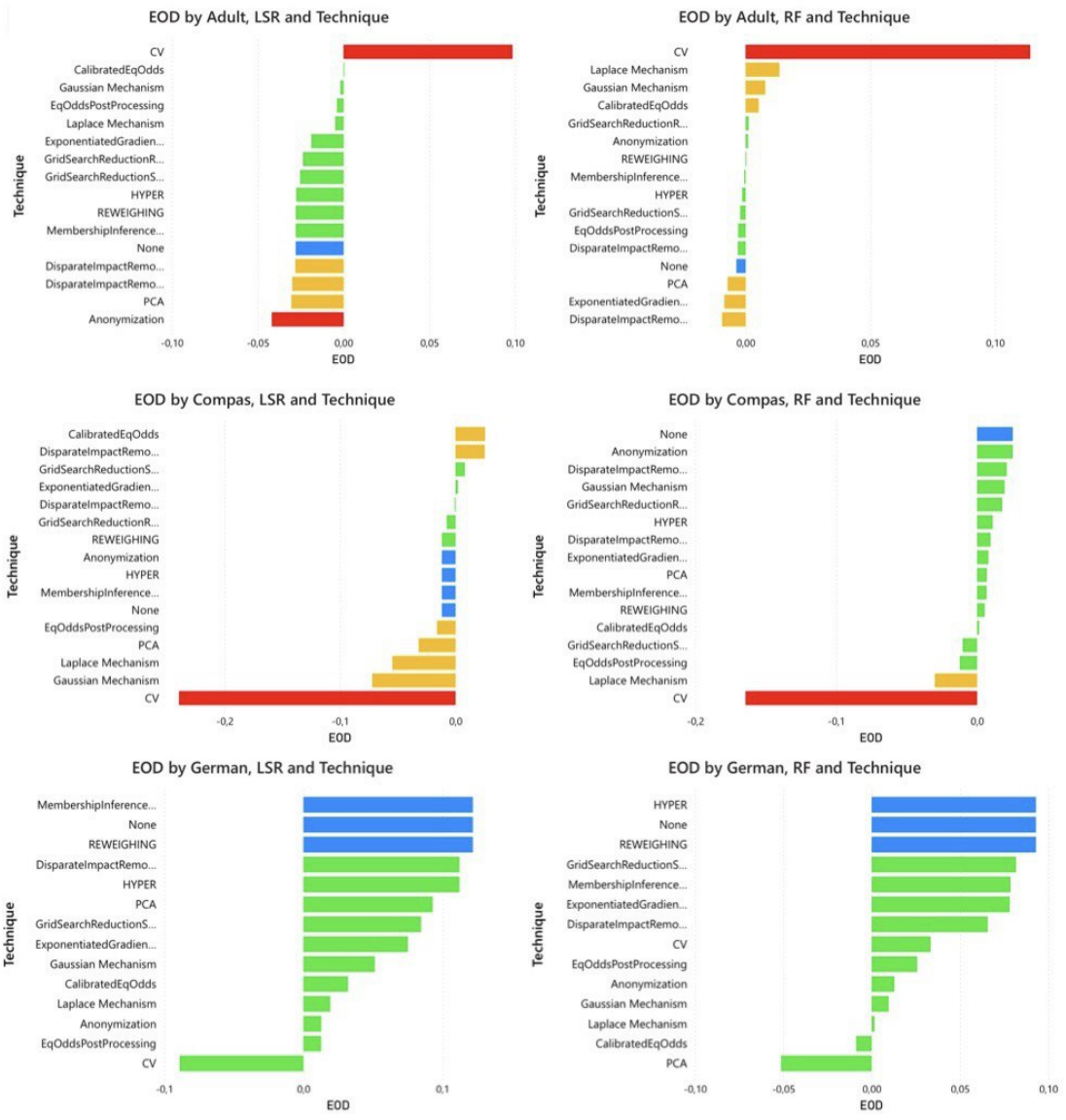

Figure 5.9: EOD results

Table 5.6: Results of the Dunn test for EOD

Subject	**Technique**	**P-value**
Adult-LSR	Cross Validation	<0,0001
Adult-LSR	EqOdds PostProcessing	0,0062
Adult-LSR	Calibrated EqOdds	0,0036
Adult-LSR	Gaussian Mechanism	0,0020
Adult-LSR	Laplace Mechanism	0,0177
Adult-RF	Cross Validation	<0,0001
Adult-RF	Laplace Mechanism	0,0013
Adult-RF	Gaussian Mechanism	0,0097
Compas-LSR	Disparate Impact Remover_Race	0,0052
Compas-LSR	Cross Validation	0,0091
Compas-RF	Cross Validation	<0,0001
German-LSR	Laplace Mechanism	<0,0001
German-LSR	EqOdds PostProcessing	<0,0001
German-LSR	Anonymization	<0,0001
German-LSR	Cross Validation	<0,0001
German-LSR	Grid Search Reduction_Sex	0,0358
German-LSR	Exponentiated Gradient Reduction	0,0067
German-LSR	Gaussian Mechanism	0,0003
German-LSR	Calibrated EqOdds	0,0002
German-RF	Anonymization	0,0074
German-RF	Laplace Mechanism	<0,0001
German-RF	EqOdds PostProcessing	0,0099
German-RF	PCA	<0,0001
German-RF	Gaussian Mechanism	0,0044
German-RF	Calibrated EqOdds	0,0022

Figure 5.10 and Table 5.7 show results for Average Odds Difference metric.

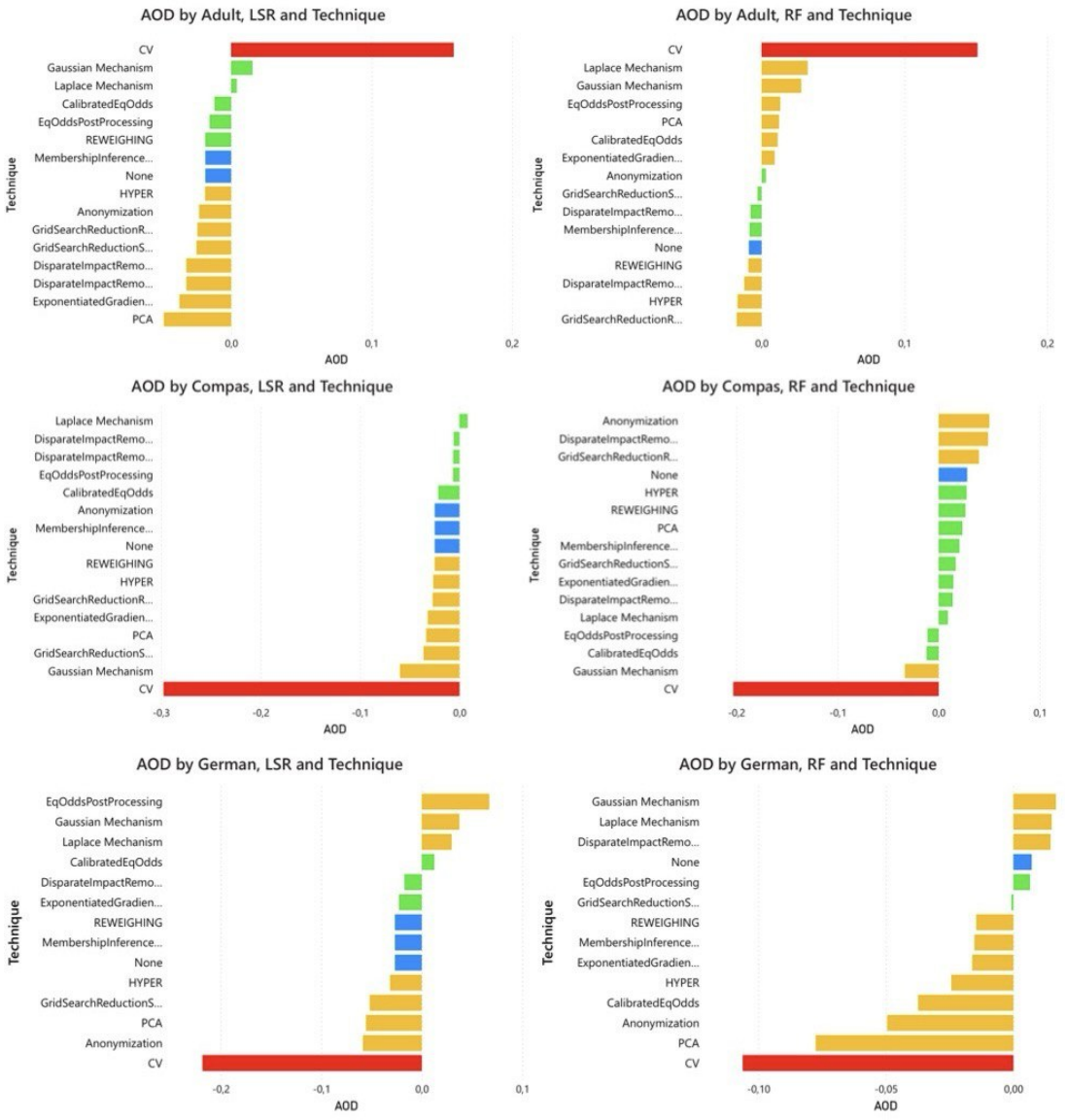

Figure 5.10: AOD results

The situation here is slightly different. There are techniques that overall have improved this metric, but just as many have made it worse. In common, Cross Validation was the worst technique as it took the AOD value out of the allowable fairness range in all six cases.

Table 5.7: Results of the Dunn test for AOD

Subject	**Technique**	**P-value**
Adult-LSR	Disparate Impact Remover_Sex	0,0177
Adult-LSR	Exponentiated Gradient Reduction	0,0166
Adult-LSR	Disparate Impact Remover_Race	0,0029
Adult-LSR	PCA	<0,0001
Adult-RF	Cross Validation	<0,0001
Adult-RF	Laplace Mechanism	0,0013
Adult-RF	Gaussian Mechanism	0,0014
Compas-LSR	Gaussian Mechanism	0,0012
Compas-LSR	Cross Validation	<0,0001
Compas-LSR	Grid Search Reduction_Sex	0,0279
Compas-RF	Cross Validation	<0,0001
Compas-RF	Gaussian Mechanism	0,0077
Compas-RF	EqOdds PostProcessing	0,0264
German-LSR	Anonymization	0,0207
German-LSR	Cross Validation	0,0017
German-RF	Anonymization	0,0063
German-RF	Cross Validation	0,0001
German-RF	PCA	0,0012

Figure 5.11 and Table 5.8 show results for Pointwise Differential Training Privacy metrics.

Overall, there is a reduction in privacy exposure, as expected due to privacy techniques and unexpectedly accuracy and post-processing techniques for bias mitigation.

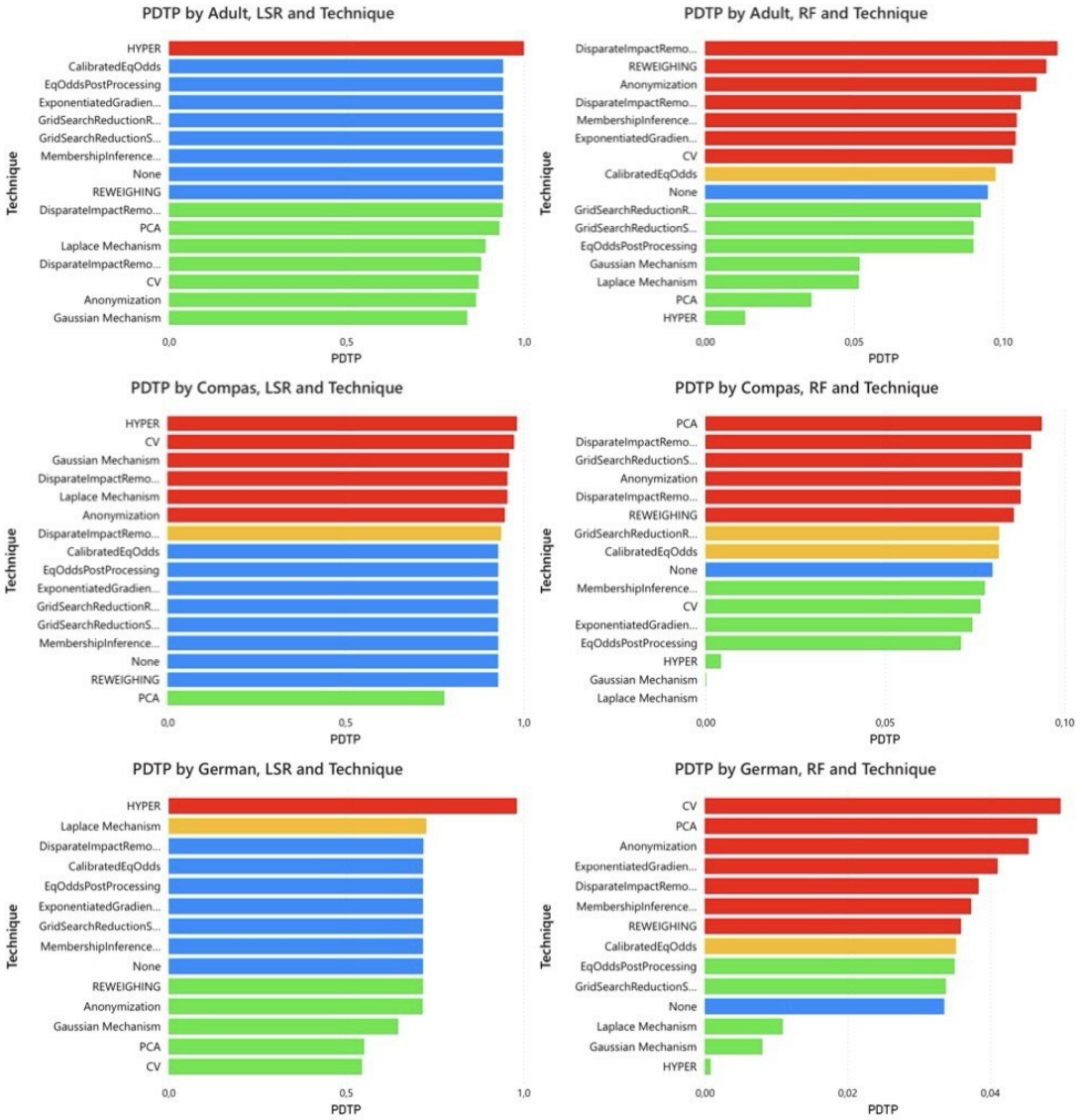

Figure 5.11: PDTP results

Table 5.8: Results of the Dunn test for PDTP

Subject	**Technique**	**P-value**
Adult-LSR	Laplace Mechanism	0,0120
Adult-LSR	Cross Validation	0,0002
Adult-LSR	Disparate Impact Remover_Race	0,0016
Adult-LSR	Anonymization	<0,0001
Adult-LSR	Gaussian Mechanism	<0,0001
Adult-RF	PCA	0,0016
Adult-RF	HyperParam Optimization	0,0002
Adult-RF	Laplace Mechanism	0,0282
Adult-RF	Gaussian Mechanism	0,0217
Compas-LSR	Gaussian Mechanism	0,0002
Compas-LSR	Cross Validation	<0,0001
Compas-LSR	HyperParam Optimization	0<0,0001
Compas-LSR	Disparate Impact Remover_Sex	0,0045
Compas-LSR	Laplace Mechanism	0,0045
Compas-RF	HyperParam Optimization	0,0371
Compas-RF	Gaussian Mechanism	0,0033
Compas-RF	Laplace Mechanism	0,00021
German-LSR	HyperParam Optimization	0,0053
German-LSR	Laplace Mechanism	0,0420
German-LSR	Gaussian Mechanism	0,0420
German-LSR	Cross Validation	0,0005
German-LSR	PCA	0,0053
German-RF	HyperParam Optimization	0,0417
German-RF	Cross Validation	0,0037
German-RF	PCA	0,0381

Figure 5.12 and Table 5.9 show results for SHAPr.

A marked improvement in terms of privacy risk reduction can be seen in all six cases. As could be expected, accuracy improvement techniques such as PCA and Hyper-parameters Optimization are the main ones to have worsened the metrics.

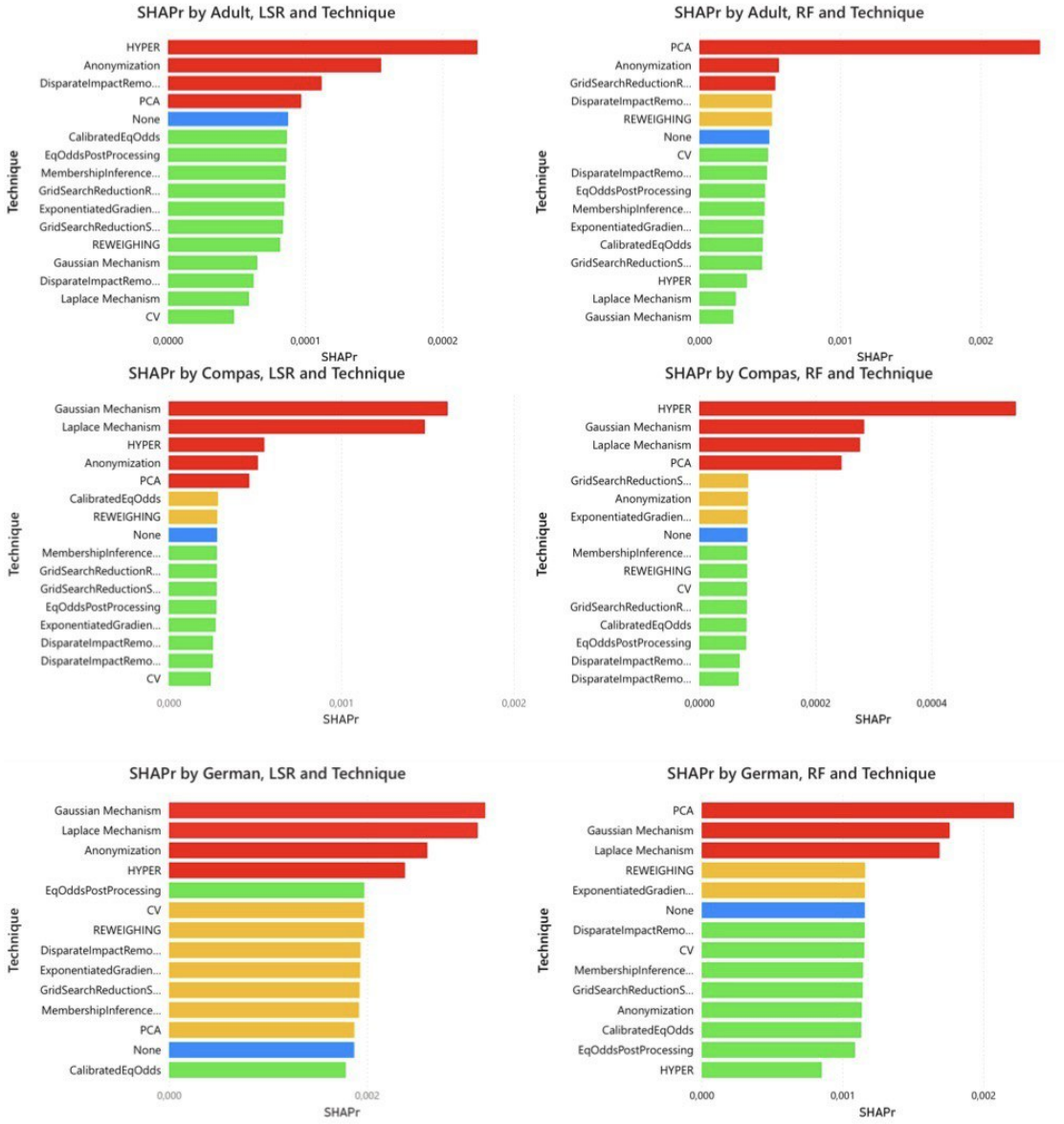

Figure 5.12: SHAPr results

Table 5.9: Results of the Dunn test for SHAPr

Subject	**Technique**	**P-value**
Adult-LSR	Laplace Mechanism	0,0117
Adult-LSR	Cross Validation	0,0006
Adult-LSR	Disparate Impact Remover_Sex	0,0187
Adult-LSR	HyperParam Optimization	0,0422
Adult-RF	HyperParam Optimization	0,0098
Adult-RF	Laplace Mechanism	0,0011
Adult-RF	Gaussian Mechanism	0,0003
Compas-LSR	Gaussian Mechanism	0,0002
Compas-LSR	HyperParam Optimization	00,0115
Compas-LSR	Laplace Mechanism	0,0011
Compas-LSR	Anonymization	0,0366
Compas-RF	HyperParam Optimization	0,0210
Compas-RF	Gaussian Mechanism	0,0049
Compas-RF	Laplace Mechanism	0,0155
Compas-RF	Disparate Impact Remover_Sex	0,0240
German-LSR	Anonymization	0,0035
German-LSR	Laplace Mechanism	$<$0,0001
German-LSR	Gaussian Mechanism	$<$0,0001
German-LSR	HyperParam Optimization	0,0351
German-RF	HyperParam Optimization	0,00080
German-RF	Laplace Mechanism	0,0482
German-RF	PCA	0,0026

Figure 5.13 and Table 5.10 show results for differential privacy loss.

It is clear from the figure that the Logistic Regression model performed better than the Random Forest. Unexpectedly, Cross Validation, PCA and bias mitigation techniques also led to improvements in four out of six cases.

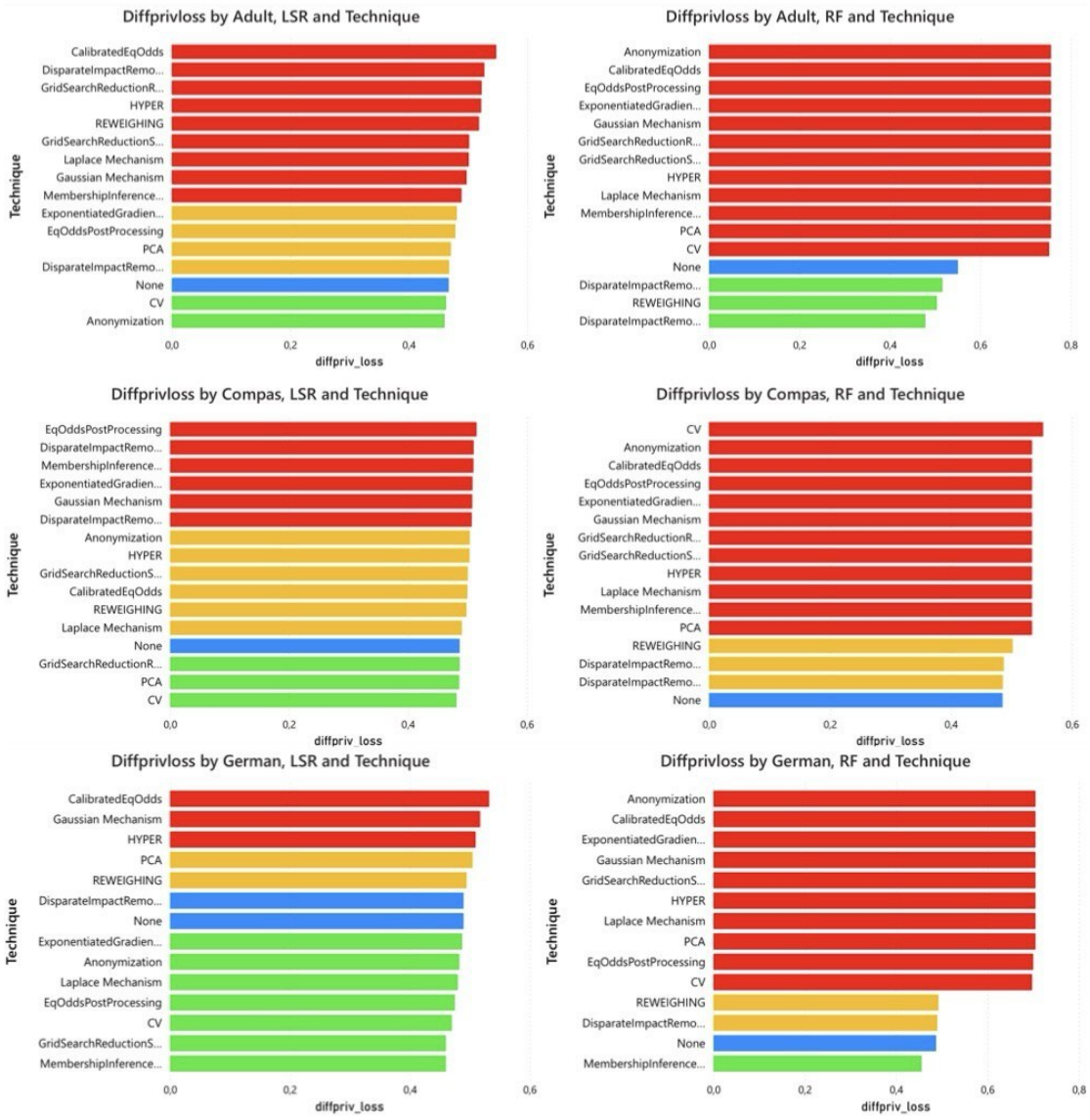

Figure 5.13: Differential privacy loss results

Table 5.10: Results of the Dunn test for differential privacy loss

Subject	**Technique**	**P-value**
Adult-RF	HyperParam Optimization	<0,0001
Adult-RF	Laplace Mechanism	<0,0001
Adult-RF	Gaussian Mechanism	<0,0001
Adult-RF	Calibrated EdOdds	<0,0001
Adult-RF	Anonymization	<0,0001
Adult-RF	EqOdds PostProcessing	<0,0001
Adult-RF	Exponentiated Gradient Reduction	<0,0001
Adult-RF	Grid Search Reduction_Race	<0,0001
Adult-RF	Grid Search Reduction_Sex	<0,0001
Adult-RF	Membership Inference Attack	<0,0001
Adult-RF	PCA	<0,0001
Compas-RF	Cross Validation	<0,0001
Compas-RF	Gaussian Mechanism	0,0478
German-RF	Exponentiated Gradient Reduction	<0,0001
German-RF	Calibrated EqOdds	<0,0001
German-RF	Laplace Mechanism	<0,0001
German-RF	PCA	<0,0001
German-RF	Gaussian Mechanism	<0,0001
German-RF	Anonymization	<0,0001
German-RF	HyperParam Optimization	0,0004
German-RF	Grid Search Reduction_Sex	0,0015
German-RF	EqOdds PostProcessing	0,0201

5.2.2 Effect of techniques on responses

Effect of accuracy techniques on accuracy

Figure 5.14 shows line graphs illustrating for each accuracy metric the performance of the accuracy techniques according to the three study datasets.

It can be observed that overall PCA had a negative effect.

Hyper-parameters tuning is the best technique for balanced accuracy and f1 score.

For precision metric, no technique brought an improvement to the German dataset. For the other two datasets Cross Validation and Hyper-parameters Optimization gave an improvement. The same evaluations apply to recall.

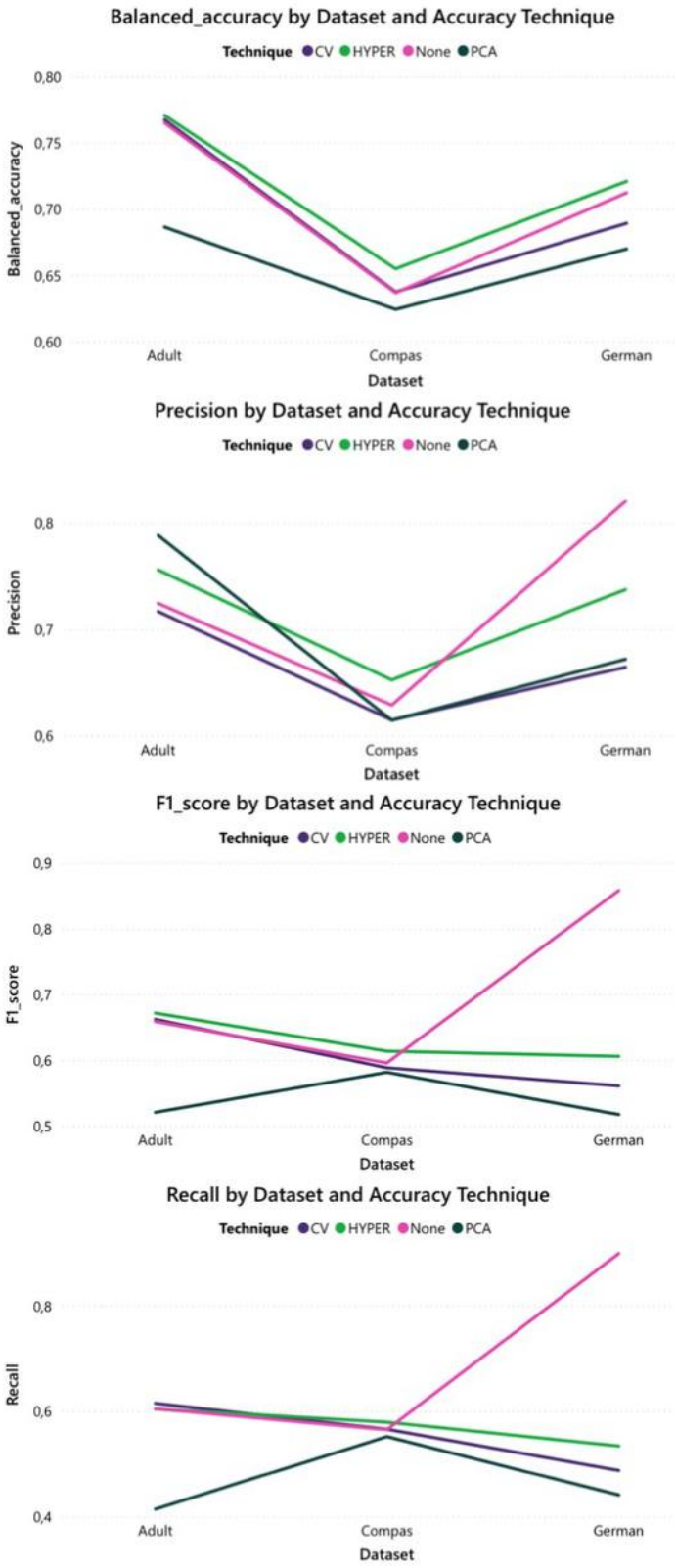

Figure 5.14: Accuracy techniques on accuracy

Effect of fairness techniques on fairness

Figure 5.15 shows scatter plots illustrating for each fairness metric the performance of fairness techniques based on the three study datasets.

It should be reminded that the goal is to reach values of 0 to indicate total fairness.

It can be seen that for all three fairness metrics, the application of the techniques led to an improvement in values.

For the SPD metric, Reweighing and Exponentiated Gradient Reduction proved to be the best techniques for the German and Compas datasets. For the Adult dataset, Calibrated EqOdds and EqOdds Postprocessing reached values close to zero, while they were worse in the other two cases.

For the EOD metric, all techniques made improvements, in particular Calibrated EqOdds.

For the AOD metric, Calibrated EqOdds and EqOdds Postprocessing reached ideal fairness values for the Adult dataset. In the case of Compas, Reweighing was the best technique. Disparate Impact Remover was the worst technique for Adult and Compas, but the best for German.

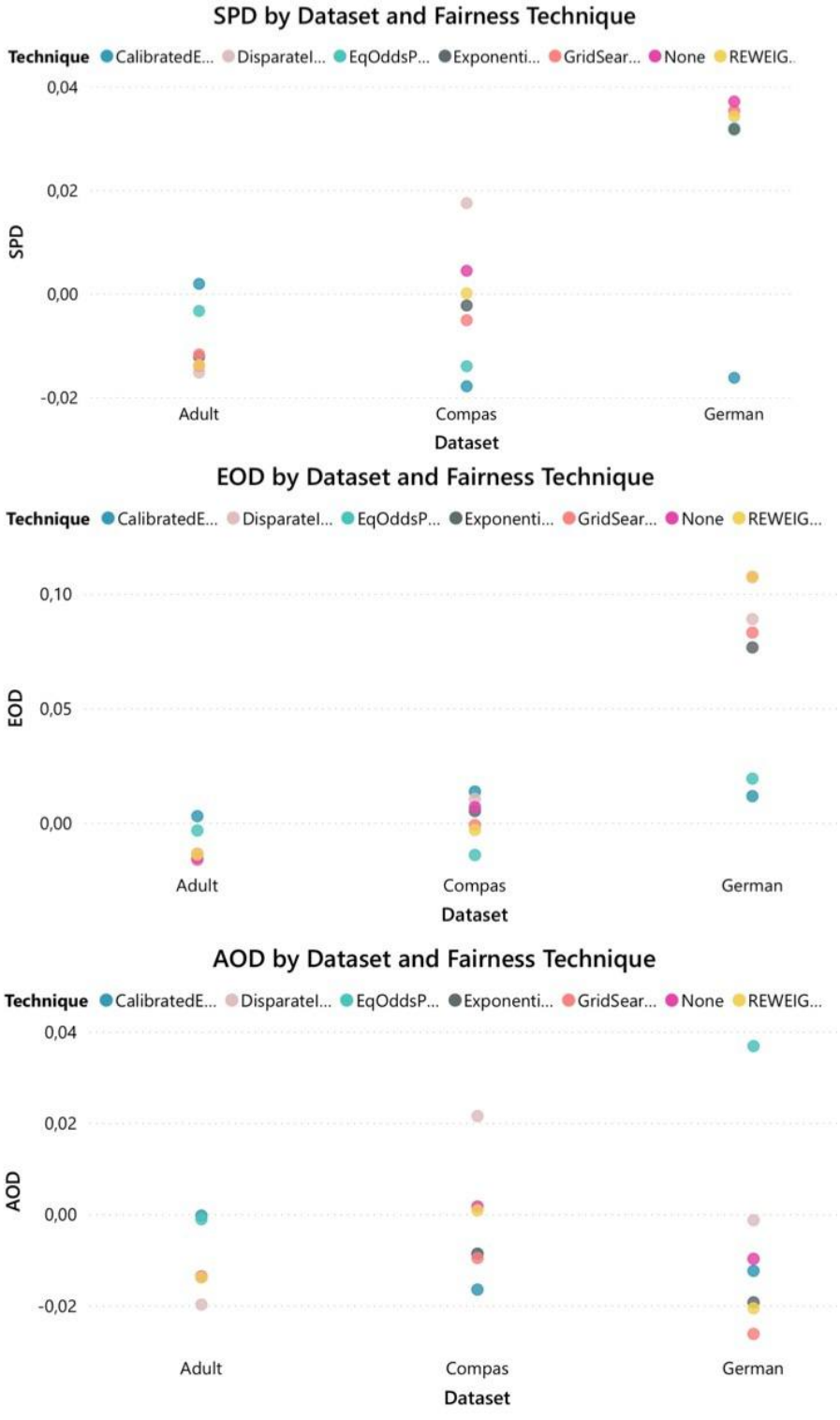

Figure 5.15: Fairness techniques on fairness

Effect of privacy techniques on privacy

Figure 5.16 shows line graphs illustrating for each privacy metric the performance of privacy techniques according to the three datasets.

As expected, SHAPr is reduced by Membership Inference Attack. Laplace Mechanism also reduced this metric for Adult dataset.

All techniques improve PDTP, with Gaussian Mechanism doing particularly well. Anonymization performed worse for Compas and German datasets.

Membership Inference Attack is the most effective for German dataset in DiffPrivLoss. No one technique improved in the other cases.

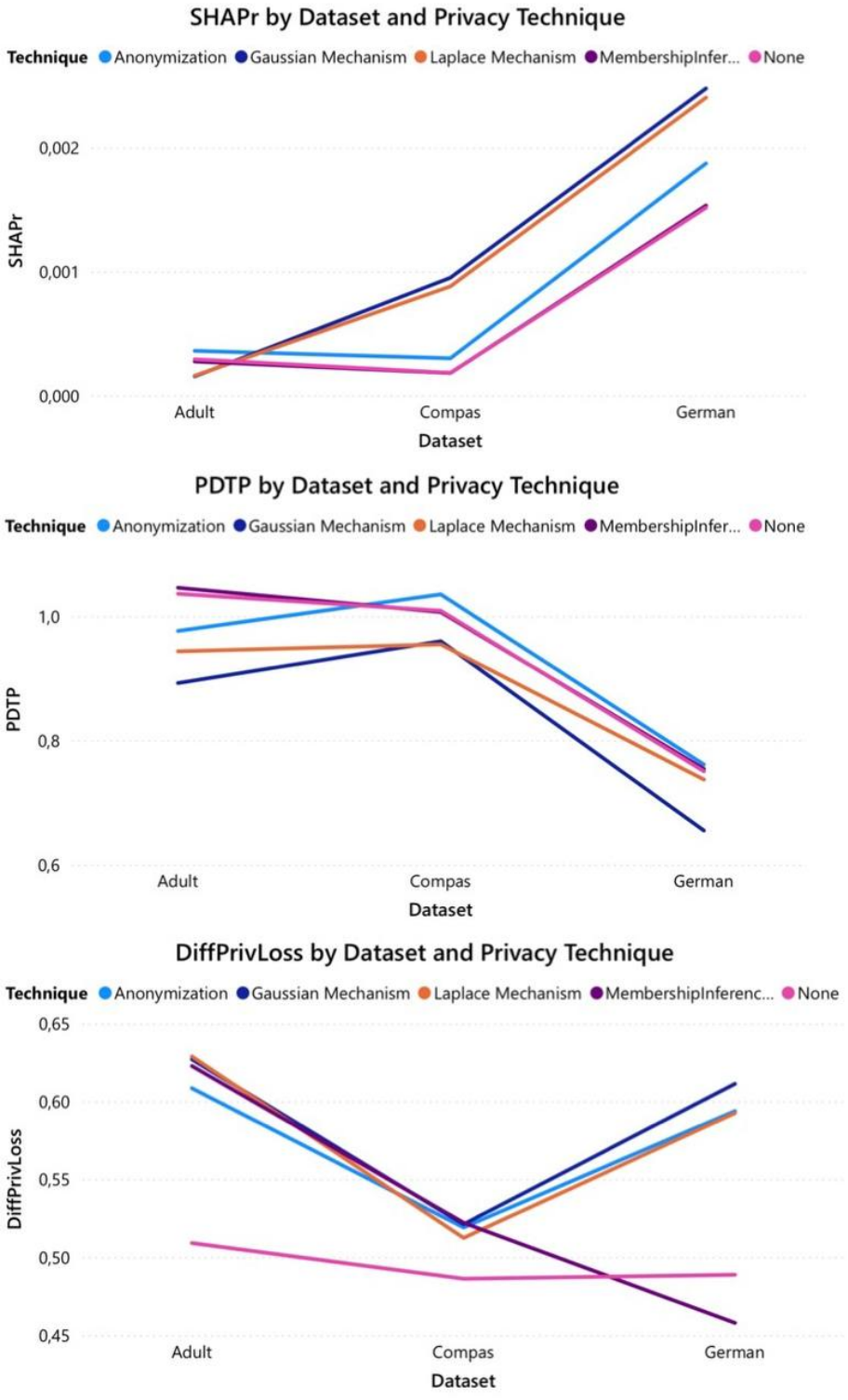

Figure 5.16: Privacy techniques on privacy

5.2.3 Cross-Effect of techniques on responses

Effect of accuracy techniques on fairness

Figure 5.17 shows scatter plots illustrating for each fairness metric the performance of accuracy techniques based on the three study datasets.

It can be observed how unexpectedly for all three metrics PCA had a positive impact reaching values close to zero, followed by Hyper-parameters Optimization. Cross-Validation is the worst technique, reaching values that exceed the fairness range.

Figure 5.17: Accuracy techniques on fairness

Effect of accuracy techniques on privacy

Figure 5.18 shows strip charts illustrating for each privacy metric the performance of accuracy techniques according to the three study datasets.

For SHAPr, Cross-Validation is the best technique. PCA is the worst one.

For PDTP, PCA is the best technique for Adult and Compas datasets. Cross-Validation has a positive impact, except for Compas dataset for which it exceed the baseline value.

For DiffPrivLoss, as expected, all accuracy improvement techniques had a negative impact, Hyper-parameter Optimization particularly.

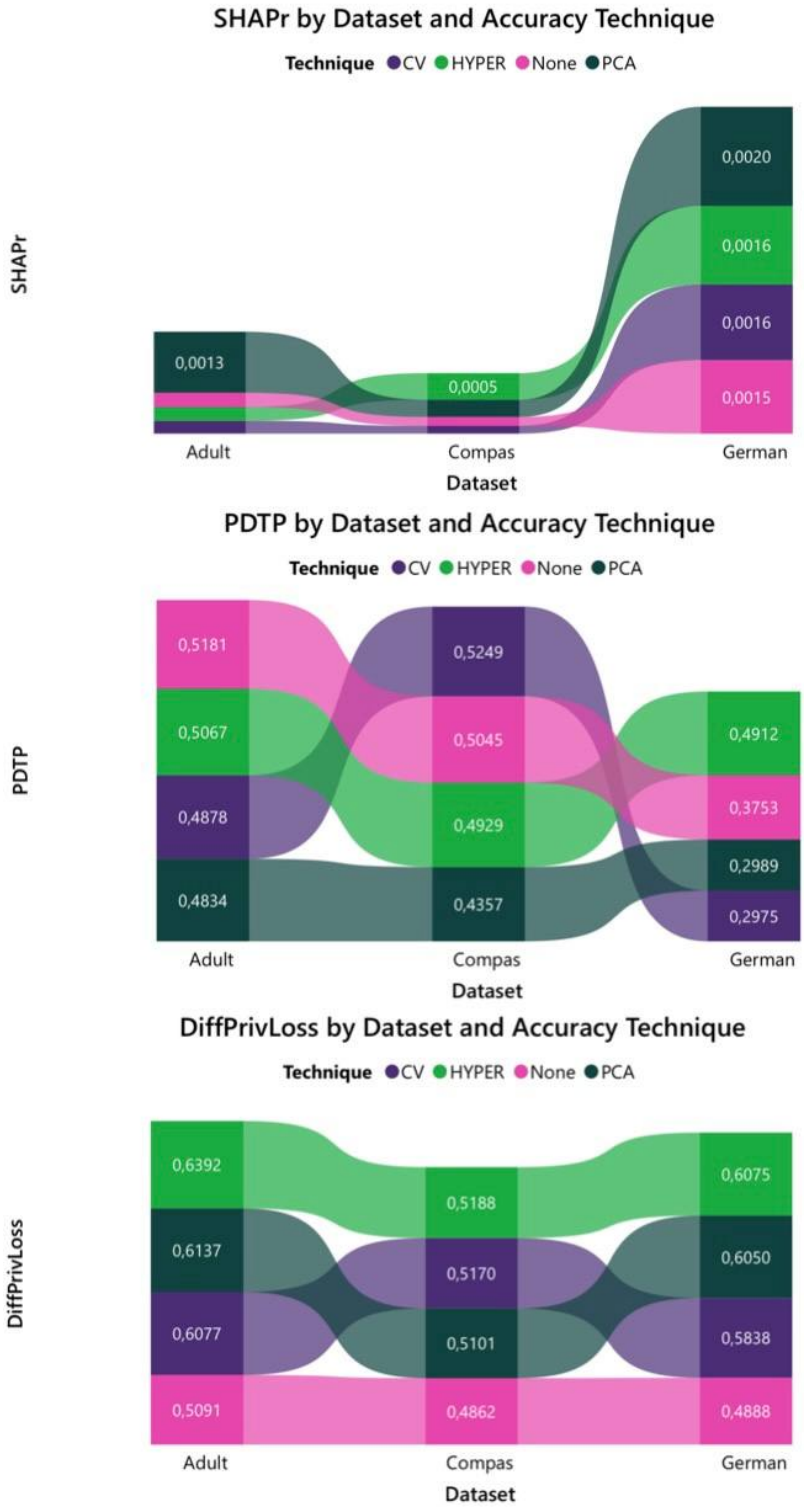

Figure 5.18: Accuracy techniques on privacy

Effect of fairness techniques on accuracy

Figure 5.19 shows strip charts illustrating for each accuracy metric the performance of fairness techniques based on the three study datasets.

For balanced accuracy, unexpectedly, also Reweighing and Calibrated EqOdds had a positive impact to Compas dataset. Grid Search Reduction and Exponentiated Gradient Reduction had a positive impact to German dataset.

For Precision, only Compas dataset had improvements. In this case, Reweighing outperformed.

For recall, Disparate Impact Remover is the best technique for Adult and Compas datasets. No one technique led improvements to German dataset.

For f1 score, German dataset had no improvements. Disparate Impact Remover had a positive impact to Adult dataset, while Calibrated EqOdds is the best technique for Compas dataset.

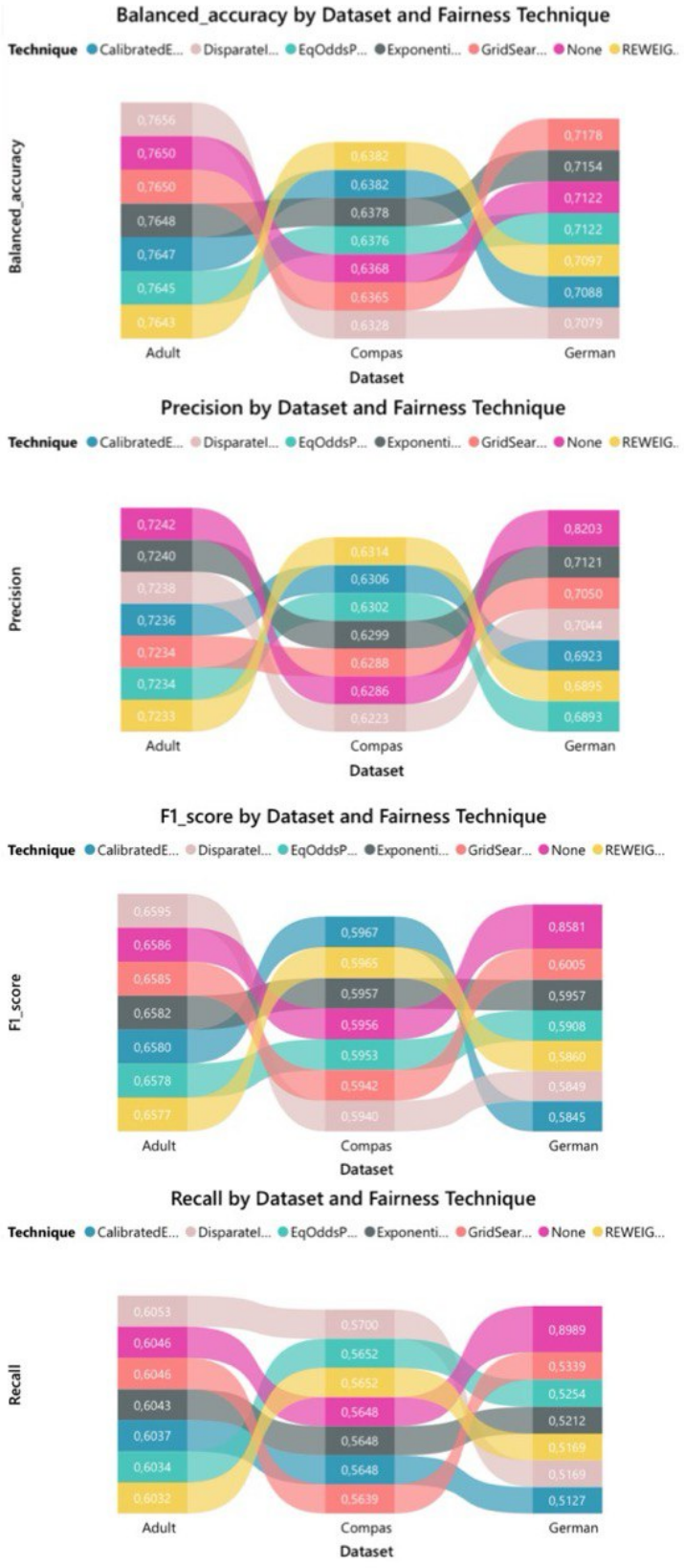

Figure 5.19: Fairness techniques on accuracy

Effect of fairness techniques on privacy

Figure 5.20 shows strip charts illustrating for each privacy metric the performance of fairness techniques based on the three study datasets.

It can be observed that the Disparate Impact Remover is the worst technique for PDTP while the best for the DiffPrivLoss. Instead, Calibrated EqOdds is the best for the SHAPr, while the worst for the DiffPrivLoss.

For SHAPr, Reweighing is the worst technique.

For PDTP, EqOdds Postprocessing led to improvements to Adult and Compas datasets.

For Diffprivloss, only the Adult dataset had an improvement with Disparate Impact Remover.

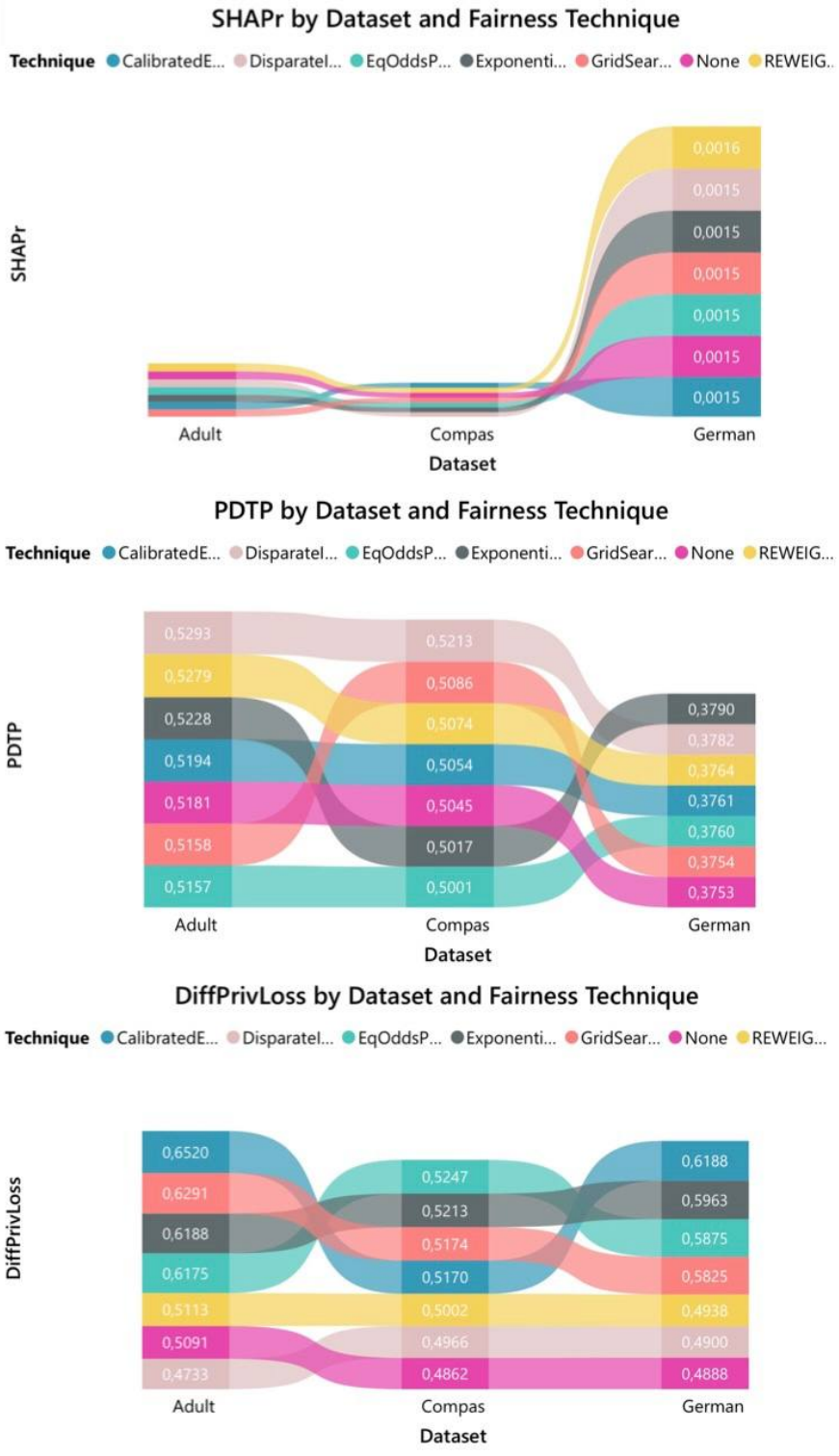

Figure 5.20: Fairness techniques on privacy

Effect of privacy techniques applied to accuracy

Figure 5.21 shows strip charts illustrating for each accuracy metric the performance of privacy techniques according to the three study datasets.

It can be seen that Anonymization is the best for balanced accuracy and recall while the worst for precision in the case of the Adult dataset.

For balanced accuracy and precision, the Membership Inference Attack performs well.

The Gaussian Mechanism turns out to be the worst technique for balanced accuracy, f1 score and recall.

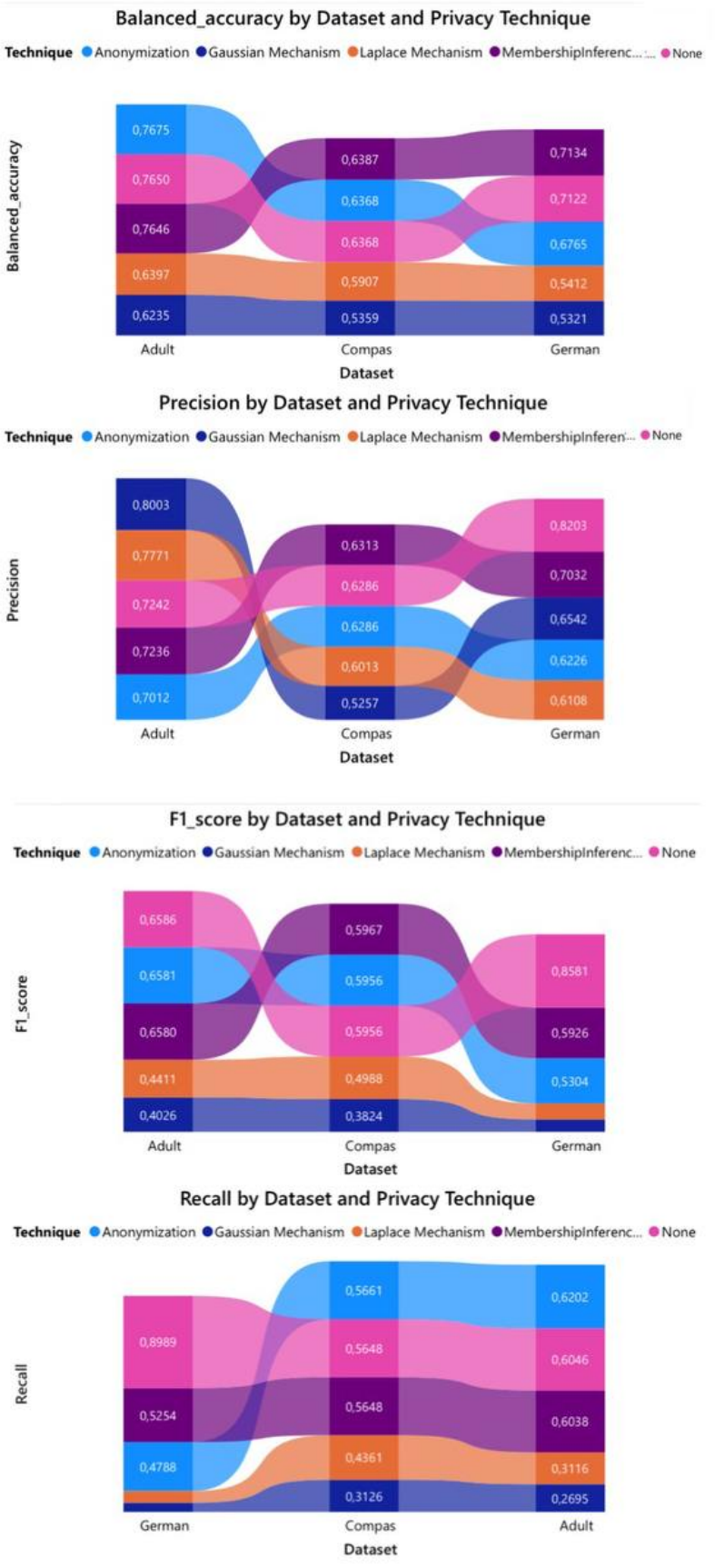

Figure 5.21: Privacy techniques on accuracy

Effect of privacy techniques on fairness

Figure 5.22 shows scatter plots illustrating for each fairness metric the performance of privacy techniques across the three study datasets.

For SPD, Anonymization turns out to be the worst technique. The others improve on the baseline results, Membership Inference Attack and Laplace Mechanism particularly.

For EOD, Laplace Mechanism is also the best technique for Adult and German datasets. Membership Inference Attack reaches the ideal fairness value for Compas dataset. There is no one technique worse than the baseline.

For AOD, the worst technique is the Gaussian Mechanism. Anonymization has a positive impact to Adult dataset. Membership Inference Attack reaches the ideal fairness value for Compas dataset.

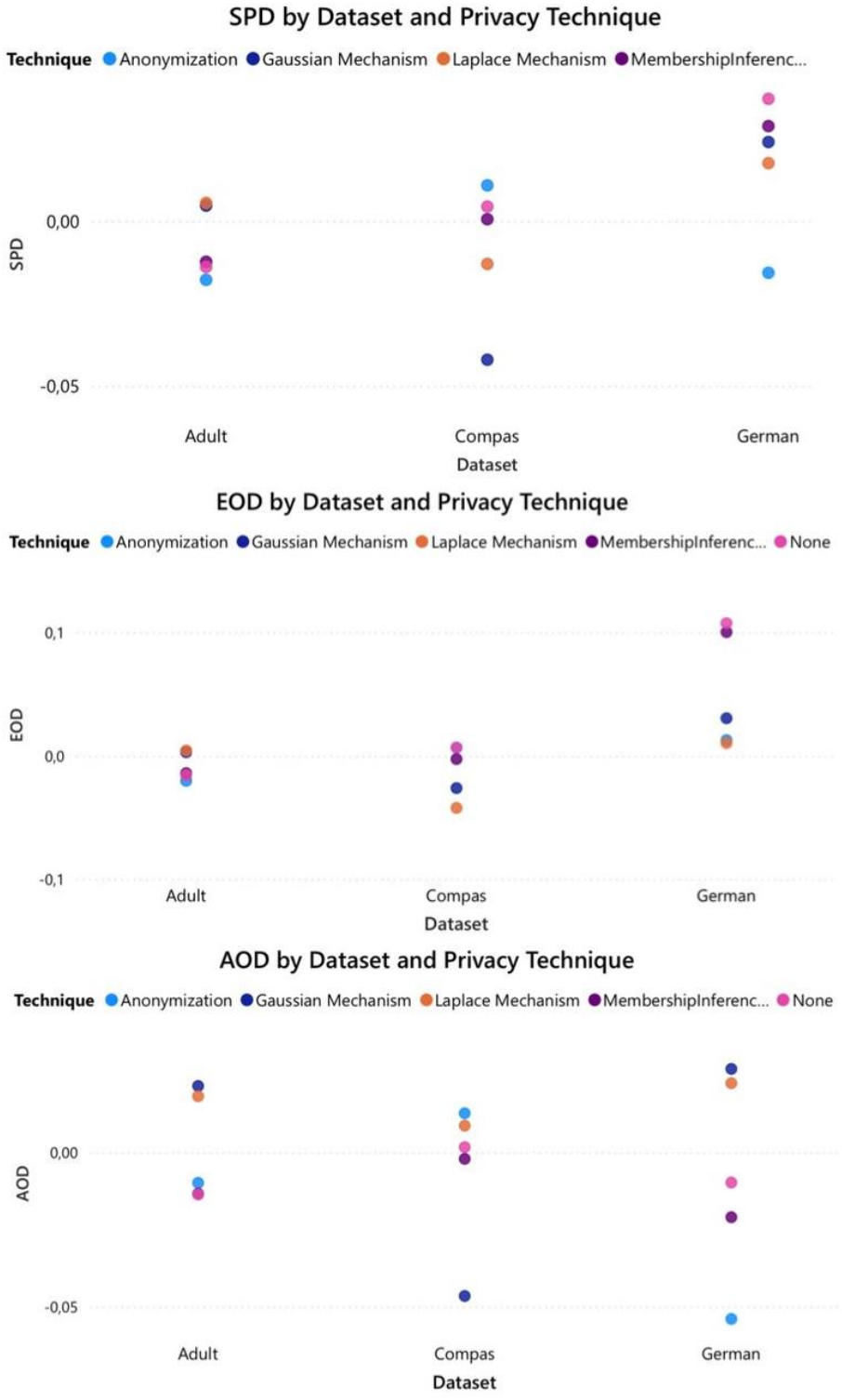

Figure 5.22: Privacy techniques on fairness

5.2.4 Effect of all techniques on all responses

A MANOVA (Multivariate ANalysis Of Variance) test was performed. It analysed if significant differences existed between the groups of the independent variable (technique) and several dependent variables (all response variables) [53].

Wilks' test is one of the most common Manova tests. It was performed with *statsmodel* in Python. If the test is significant, it indicates that at least one of the independent variables has a significant impact on the dependent variables. Wilks' lambda value of 0.1334 indicated that the model explains most of the variance, as the value is close to 0. After calculating Wilks' Lambda, a significance test such as the F-test is performed to see whether the value of Wilks' Lambda is statistically significant (p-value < 0.05). The Technique is highly significant ($p < 0.0001$), which means that there is a significant variance explained by the overall model, i.e. the response variables are not all equal.

The aim was to visualise the effects of these results in a certain way. One possibility consists of a scatter plot showing the separation of the techniques based on the canonical scores, projections of the response variables into a new space, where the correlation between the dependent variables (**response variables**) and the independent variable (**techniques**) is maximized.

The statsmodels library does not directly support the extraction of

canonical scores; however, these can be derived using Scikit-Learn for **Canonical Correlation Analysis (CCA)**. The maximum number of canonical components that CCA can generate is the lesser of the number of response variables (10) and techniques (16) minus one. In this case, the maximum number is 8. Using the **canonical correlation coefficients**, shown in Table 5.11, one can assess how well two or more canonical components are associated. A negative canonical correlation does not necessarily mean that the techniques have a negative impact on the response variables, but rather that the direction of change in the response variables associated with the techniques is inverse to that expected. CCA with 4 components was chosen.

Table 5.11: Canonical correlation coefficients

Component	**Value**
9	-0.2039
8	-0.1059
7	-0.0250
6	0.0614
5	0.0766
4	0.0789
3	0.0743
2	0.0631

The correlation matrix in Figure 5.23 shows how all response vari-

ables are interrelated and the weight of each variable in each canonical component. It can be noticed that the first component is mainly represented by accuracy metrics, the second and fourth by fairness metrics, and the third by privacy metrics.

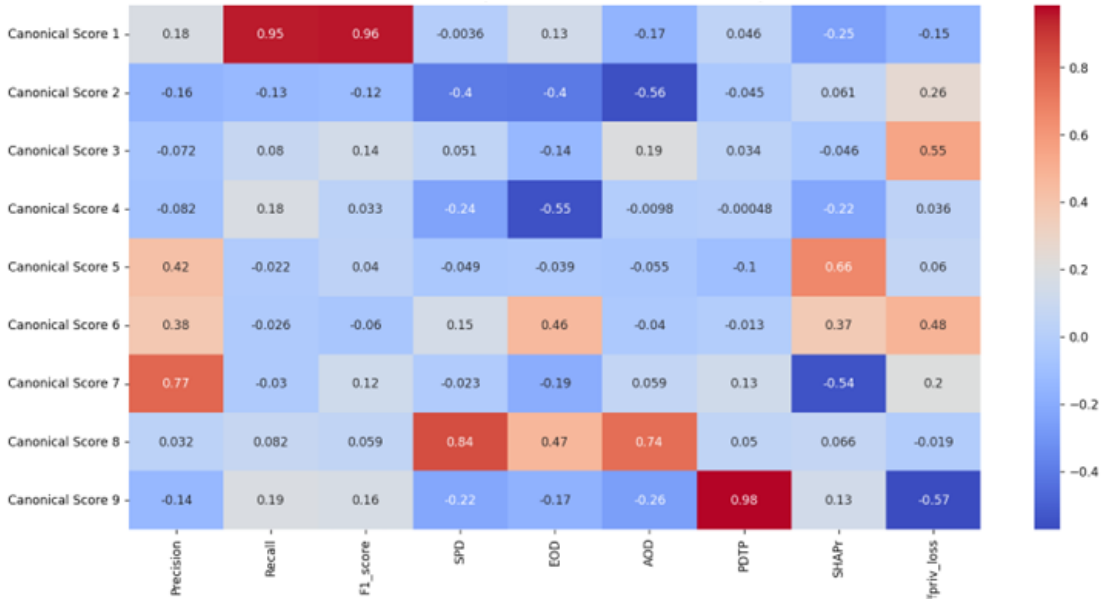

Figure 5.23: Correlation matrix

Figure 5.24 shows the pair-plots of the components. The distance between the points on the graph reflects how much the techniques differ for the combination of response variables. Points close together along an axis indicate that the corresponding techniques have a similar effect concerning that combination of response variables. Distant points along the same axis indicate that the techniques have different effects. It can also be interpreted according to the direction of the dots to the axes. For example, techniques that lie high along the y-axis indicate that they affect the second linear combination of response variables

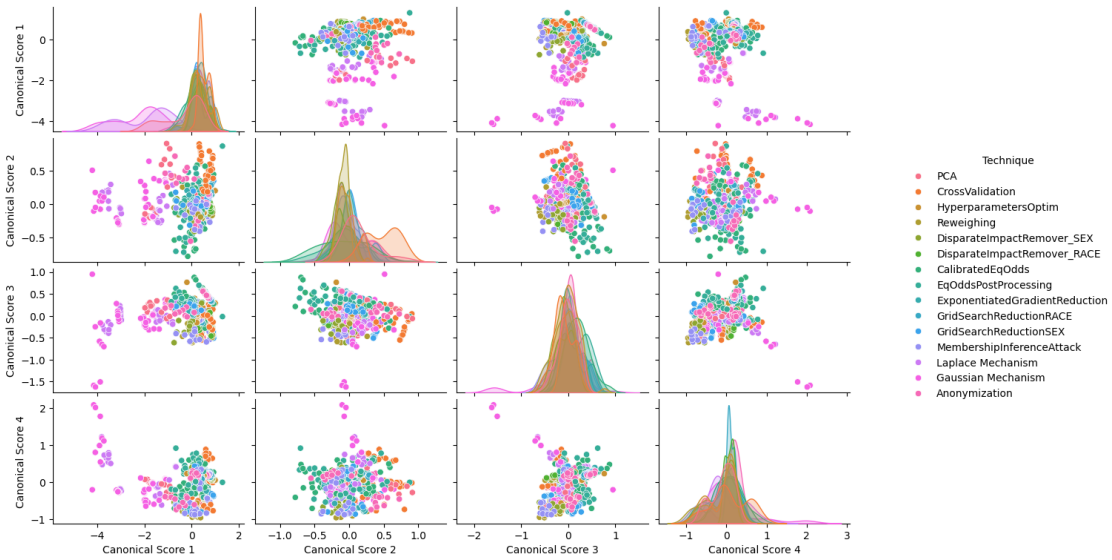

Figure 5.24: Canonical Correlation Analysis

the most.

It can be noted in Figure 5.24 that in the case of the first canonical component, Laplace and Gaussian mechanisms had different effects with respect to the other techniques; indeed, these two techniques worsened the precision, recall, and F1 score values. Looking at the second component, it is possible to see how the Cross Validation differs from the other techniques; indeed, this technique had a negative impact on fairness metrics. Looking at the fourth component, it can be observed how the Gaussian mechanism differed from the other techniques.

Summary of findings

- Random Forest performs better than Logistic Regression in accuracy and fairness metrics.
- Logistic Regression outperforms Random Forest in Diff-PrivLoss.
- German dataset has more deterioration than improvement in accuracy metrics compared to the other two datasets.
- Accuracy metrics improved with unexpected gains from bias mitigation and privacy techniques.
- Fairness metrics improved, especially with Random Forest, although Cross Validation worsened AOD.
- Bias mitigation techniques brings unexpected improvements in privacy metrics.
- PCA is the worst technique for all metrics.
- Disparate Impact Remover is the best bias mitigation technique that improved accuracy metrics.
- Laplace Mechanism is the best privacy risk mitigation technique that improved fairness metrics.

Chapter 6

Insights and limitations

6.1 Results overview

Figure 6.1 shows a summary of the results obtained from the experiments, highlighting the techniques that had a better and a worst impact on response variables.

It should be noted that:

- Cross Validation is a good technique for privacy but not for fairness;
- Hyper-parameters Optimization is not a good technique for privacy.
- Reweighing is good for accuracy but not for privacy.

Table 6.1: Techniques with greatest effect on...

...Accuracy	**...Fairness**	**...Privacy**
1) Reweighing	1) Hyper-Param Optim	1) EqOdds PostProcessing
2) Disparate Impact Remover	2) Anonymization	2) Cross Validation
3)Anonymization	3) Laplace Mechanism	3) Grid Search Reduction
...	...	...
13) Gaussian Mechanism	13) PCA	13) Hyper-Param Optim
14) Laplace Mechanism	14) Cross Validation	14) Disparate Impact Remover
15) Calibrated EqOdds	15) Gaussian Mechanism	15) Reweighing

Hence, in cases where a practitioner is interested mainly in finding a **trade-off between accuracy and fairness**, the application of Reweighing, Hyper-parameters Optimization, Disparate Impact Remover is recommended.

In cases where a practitioner is interested mainly in the **trade-off between accuracy and privacy** the application of Cross Validation and Anonymization.

When one is interested in the **trade-off between fairness and privacy**, the application of Laplace Mechanism, Grid Search Reduction and EqOdds Post Processing is recommended.

The techniques that are not present in the figure are those that had an average impact on the metrics and that can be applied in the case where one is interested in a **trade-off between accuracy, fairness, and privacy**.

6.2 Threats to validity

As for any experimental study, there are limitations that may affect the validity of the results.

Construct validity. The results may be different if other classification models are used. It was chosen to focus on the study of group fairness metrics. Future work may extend the study to individual fairness metrics to verify the reliability of the results. The number of repetitions of ten due to the memory limitations of the hardware system on which the experiments were conducted may have affected the robustness of the results. Future work may attempt to increase the number of repetitions.

External validity. Three datasets of different sizes were considered, which are public, well known, and widely adopted in the literature. However, this limits the applicability to other domains. Future work may extend the methodology presented to other types of datasets.

Internal validity. The developed framework was realized by integrating several toolkits, many of which were released by IBM. Although most of them come from the same company, the integration of these tools has not yet been carried out and, as a result, compatibility or synergy problems may arise between the various components. The official tutorials of the individual toolkits, which were used as a reference point during development, still present open issues in the codes, which could affect the stability or accuracy of the implementations.

Chapter 7

Conclusions

This thesis addressed the problem of joint evaluation of accuracy, fairness, and privacy in Machine Learning systems to assess the effect of improvement and mitigation techniques on quality goals individually and intersectionally.

Experiments were conducted using a set of benchmark datasets and commonly adopted classification models to analyze the effects of techniques in these dimensions.

One of the expected findings is that improvement in a single dimension, such as fairness or privacy, can often occur at the expense of other dimensions, such as accuracy. However, it was surprising to observe how some techniques to improve or mitigate one dimension unexpectedly helped to improve other response variables as well. For instance, some methodologies proposed to improve fairness were also shown to have a positive effect on privacy.

Another relevant aspect that emerged from the experimental results concerns the possibility of identifying, for each dimension, which techniques in the respective category are the most effective. This approach enabled a better understanding of which technique is best suited to maximize specific quality objectives, depending on the context.

Despite the results obtained, it was not possible to identify an optimal solution that would guarantee the simultaneous improvement of all quality dimensions. However, this thesis has provided insight into which techniques are better than others when it comes to specific trade-offs. Depending on the combination of metrics of one's interest (e.g. accuracy-fairness-privacy, accuracy-privacy, accuracy-fairness or fairness-privacy), the results showed that some techniques balance these trade-offs better than others.

Future work could involve analysing the impact of multiple techniques applied at once. This could reveal even more effective combinations of techniques than those observed individually. This could extend the suggestions made in this thesis, providing a more solid basis for the selection of techniques according to desired quality goals.

Bibliography

[1] Joymallya Chakraborty, Suvodeep Majumder, and Tim Menzies. Bias in machine learning software: Why? how? what to do? *CoRR*, abs/2105.12195, 2021.

[2] Jason Chan and Jing Wang. Hiring Preferences in Online Labor Markets: Evidence of a Female Hiring Bias. *Management Science*, 64(7):2973–2994, July 2018.

[3] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art, 2017.

[4] Teresa Bono, Karen Croxson, and Adam Giles. Algorithmic fairness in credit scoring. *Oxford Review of Economic Policy*, 37(3):585–617, 09 2021.

[5] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. Fairness through awareness. *CoRR*, abs/1104.3913, 2011.

[6] Zhenpeng Chen, Jie M. Zhang, Max Hort, Mark Harman, and Federica Sarro. Fairness testing: A comprehensive survey and analysis of trends, 2024.

[7] Ezekiel Soremekun, Mike Papadakis, Maxime Cordy, and Yves Le Traon. Software fairness: An analysis and survey, 2022.

[8] Cynthia Dwork, Christina Ilvento, Guy N. Rothblum, and Pragya Sur. Abstracting fairness: Oracles, metrics, and interpretability. *CoRR*, abs/2004.01840, 2020.

[9] Zhenpeng Chen, Jie M. Zhang, Federica Sarro, and Mark Harman. A comprehensive empirical study of bias mitigation methods for machine learning classifiers, 2023.

[10] Sumon Biswas and Hridesh Rajan. Do the machine learning models on a crowd sourced platform exhibit bias? an empirical study on model fairness. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE '20. ACM, November 2020.

[11] Karima Makhlouf, Sami Zhioua, and Catuscia Palamidessi. Survey on causal-based machine learning fairness notions. *CoRR*, abs/2010.09553, 2020.

[12] Rūta Binkytė-Sadauskienė, Karima Makhlouf, Carlos Pinzón, Sami Zhioua, and Catuscia Palamidessi. Causal discovery for fairness. *arXiv preprint arXiv:2206.06685*, 2022.

[13] Mengdi Zhang and Jun Sun. Adaptive fairness improvement based on causality analysis, 2022.

[14] Drago Plevcko and Elias Bareinboim. Causal fairness analysis. *ArXiv*, abs/2207.11385, 2022.

[15] Joymallya Chakraborty, Suvodeep Majumder, Zhe Wu, and Tim Menzies. Fairway: SE principles for building fairer software. *CoRR*, abs/2003.10354, 2020.

[16] Zhenpeng Chen, Jie M Zhang, Federica Sarro, and Mark Harman. Maat: a novel ensemble approach to addressing fairness and performance bugs for machine learning software. In *Proceedings of the 30th ACM joint european software engineering conference and symposium on the foundations of software engineering*, pages 1122–1134, 2022.

[17] Zhenpeng Chen, Jie M. Zhang, Federica Sarro, and Mark Harman. Fairness improvement with multiple protected attributes: How far are we?, 2024.

[18] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[19] Lea Demelius, Roman Kern, and Andreas Trügler. Recent advances of differential privacy in centralized deep learning: A systematic survey, 2023.

[20] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. What can we learn privately? *CoRR*, abs/0803.0924, 2008.

[21] John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. Local privacy and statistical minimax rates. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 429–438, 2013.

[22] Héber Hwang Arcolezi, Jean-François Couchot, Bechara al Bouna, and Xiaokui Xiao. Improving the utility of locally differentially private protocols for longitudinal and multidimensional frequency estimates. *CoRR*, abs/2111.04636, 2021.

[23] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher,

Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine learning in python. *CoRR*, abs/1201.0490, 2012.

[24] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.

[25] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[26] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler

Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

[27] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[28] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. *CoRR*, abs/1912.01703, 2019.

[29] François Chollet et al. Keras. `https://keras.io`, 2015.

[30] Payam Refaeilzadeh, Lei Tang, and Huan Liu. *Cross-Validation*, pages 532–538. Springer US, Boston, MA, 2009.

[31] Huan Liu. *Feature Selection*, pages 402–406. Springer US, Boston, MA, 2010.

[32] Tong Yu and Hong Zhu. Hyper-parameter optimization: A review of algorithms and applications. *CoRR*, abs/2003.05689, 2020.

[33] Sainyam Galhotra, Yuriy Brun, and Alexandra Meliou. Fairness testing: Testing software for discrimination. *CoRR*, abs/1709.03221, 2017.

[34] Sakshi Udeshi, Pryanshu Arora, and Sudipta Chattopadhyay. Automated directed fairness testing. *CoRR*, abs/1807.00468, 2018.

[35] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33:1 – 33, 2011.

[36] Maria-Irina Nicolae, Mathieu Sinn, Tran Ngoc Minh, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Ian M. Molloy, and Benjamin Edwards. Adversarial robustness toolbox v0.2.2. *CoRR*, abs/1807.01069, 2018.

[37] Drago Plecko and Elias Bareinboim. Causal fairness analysis, 2022.

[38] Vasisht Duddu, Sebastian Szyller, and N. Asokan. Shapr: An efficient and versatile membership privacy risk metric for machine learning. *CoRR*, abs/2112.02230, 2021.

[39] Yunhui Long, Vincent Bindschaedler, and Carl A. Gunter. Towards measuring membership privacy. *CoRR*, abs/1712.09136, 2017.

[40] Naoise Holohan, Stefano Braghin, Pól Mac Aonghusa, and Killian Levacher. Diffprivlib: The IBM differential privacy library. *CoRR*, abs/1907.02444, 2019.

[41] Cynthia Dwork. Differential privacy. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *Automata, Languages and Programming*, pages 1–12, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[42] Abigail Goldsteen, Gilad Ezov, Ron Shmelkin, Micha Moffie, and Ariel Farkash. Anonymizing machine learning models. *CoRR*, abs/2007.13086, 2020.

[43] Reza Shokri, Marco Stronati, and Vitaly Shmatikov. Membership inference attacks against machine learning models. *CoRR*, abs/1610.05820, 2016.

[44] Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

[45] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Propublica compas recidivism risk score data and analysis, 2016. Accessed: 2024-04-20.

[46] Hans Hofmann. Statlog (German Credit Data). UCI Machine Learning Repository, 1994. DOI: https://doi.org/10.24432/C5NC77.

[47] Michael P LaValley. Logistic regression. *Circulation*, 117(18):2395–2399, 2008.

[48] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.

[49] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In Peter A. Flach, Tijl De Bie, and Nello Cristianini, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 35–50, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[50] Alessandro Fabris, Stefano Messina, Gianmaria Silvello, and Gian Antonio Susto. Algorithmic fairness datasets: the story so far. *Data Mining and Knowledge Discovery*, 36(6):2074–2152, September 2022.

[51] Jonathan R. Gair, Archisman Ghosh, Rachel Gray, Daniel E. Holz, Simone Mastrogiovanni, Suvodip Mukherjee, Antonella

Palmese, Nicola Tamanini, Tessa Baker, Freija Beirnaert, Maciej Bilicki, Hsin-Yu Chen, Gergely Dálya, Jose Maria Ezquiaga, Will M. Farr, Maya Fishbach, Juan Garcia-Bellido, Tathagata Ghosh, Hsiang-Yu Huang, Christos Karathanasis, Konstantin Leyde, Ignacio Magaña Hernandez, Johannes Noller, Gregoire Pierra, Peter Raffai, Antonio Enea Romano, Monica Seglar-Arroyo, Danièle A. Steer, Cezary Turski, Maria Paola Vaccaro, and Sergio Andrés Vallejo-Peña. The hitchhiker's guide to the galaxy catalog approach for dark siren gravitational-wave cosmology. *The Astronomical Journal*, 166(1):22, June 2023.

[52] R. Jain. *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling.* Wiley, 1991.

[53] D. Iacobucci. *Analysis of Variance, Experimental Design, and Multivariate ANOVA, 3e.* Earlie Lite Books, Incorporated, 2024.

Ringraziamenti

Ai miei Relatori, per avermi guidato nella mia crescita accademica dalla triennale alla magistrale credendo nelle mie potenzialità.

A mia Madre. Grazie all'amore con cui mi hai cresciuta non smetto mai di guardare il mondo con gli occhi di una bambina desiderosa di apprendere cose nuove e di catapultarmi in sfide per mettere alla prova le mie capacità e la mia determinazione. Questa è stata una delle grandi sfide negli ultimi anni; non poche volte ho pensato di trovarmi nel posto sbagliato, ma oggi posso dire di avercela fatta. Sei il motore di ogni mio traguardo.

A mia nonna, la mia seconda madre. Sei il mio scudo alle ingiustizie e alle asperità. La tua casa è il cuore pulsante della mia vita.

A Peppe. Sei un punto di riferimento essenziale.

A Nunzio. Sono grata alla vita perché tu sia presente in un giorno così importante per me. Sei entrato a farne parte delicatamente e giorno dopo giorno aggiungi e colori tasselli di emozioni e di esperienze insieme. Sei tutte le cose belle che possono capitare a una persona: fiducia, rassicurazione, tranquillità, motivazione, disponibilità, dolcezza, complicità, felicità e amore. Grazie.

A Doriana. Sei la scoperta di questa magistrale. In una classe di persone competitive, indifferenti ed egoiste avevo un po' perso la speranza di incontrare qualcuno che rendesse *pink* il mio percorso universitario, che gioisse prima di me per i miei successi e le cose belle che mi capitassero e con cui costruire un rapporto di amicizia e collaborazione sana. Gli sguardi di intesa, le risate rumorose, tu che asciughi le mie lacrime e io che freno le tue azioni impulsive. Stringersi la mano prima di sostenere ogni esame; abbracciarsi dopo la convalida. Noi, così simili e così diverse allo stesso tempo. Anche se abbiamo gusti e ritmi circadiani diversi, hai la capacità di rasserenarmi e di credere nelle mie idee. Questo obiettivo in comune ci ha unito più che mai nell'ultimo anno e sono felice che oggi condividiamo questo traguardo, finalmente ripagate di tutti i nostri sforzi. Solo noi conosciamo, l'una per l'altra, ogni singolo dettaglio e pensiero riguardante le nostre giornate di studio.