Universidade Estadual de Campinas
Instituto de Computação

Beatriz Cardoso Nascimento

A Fair Approach for Mixture of Probabilistic PCA

Uma Abordagem Justa para Mistura de PCA Probabilístico

CAMPINAS
2025

Beatriz Cardoso Nascimento

A Fair Approach for Mixture of Probabilistic PCA

Uma Abordagem Justa para Mistura de PCA Probabilístico

Dissertação apresentada ao Instituto de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Mestra em Ciência da Computação.

Dissertation presented to the Institute of Computing of the University of Campinas in partial fulfillment of the requirements for the degree of Master in Computer Science.

Supervisor/Orientador: Prof. Dr. Marcos Medeiros Raimundo
Co-supervisors/ Coorientadores: Prof. Dr. Alessandro Gaio Chimenton e Prof. Dr. Leonardo Tomazeli Duarte

Este exemplar corresponde à versão final da Dissertação defendida por Beatriz Cardoso Nascimento e orientada pelo Prof. Dr. Marcos Medeiros Raimundo.

CAMPINAS
2025

Ficha catalográfica
Universidade Estadual de Campinas (UNICAMP)
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

N17f

Nascimento, Beatriz Cardoso, 1998-
A fair approach for mixture of probabilistic PCA / Beatriz Cardoso Nascimento. – Campinas, SP : [s.n.], 2025.

Orientador: Marcos Medeiros Raimundo.
Coorientadores: Alessandro Gaio Chimenton, Leonardo Tomazeli Duarte.
Dissertação (mestrado) – Universidade Estadual de Campinas (UNICAMP), Instituto de Computação.

1. Aprendizado de máquina. 2. Redução de dimensionalidade (Estatística). 3. Aprendizado de representação. I. Raimundo, Marcos Medeiros, 1988-. II. Chimenton, Alessandro Gaio. III. Duarte, Leandro Tomazeli. IV. Universidade Estadual de Campinas (UNICAMP). Instituto de Computação. V. Título.

Informações complementares

Título em outro idioma: Uma abordagem justa para mistura de PCA probabilístico
Palavras-chave em inglês:
Machine learning
Dimension reduction (Statistics)
Representation learning
Área de concentração: Ciência da Computação
Titulação: Mestra em Ciência da Computação
Banca examinadora:
Marcos Medeiros Raimundo [Orientador]
Lehilton Lelis Chaves Pedrosa
Guilherme Dean Pelegrina
Data de defesa: 30-04-2025
Programa de Pós-Graduação: Ciência da Computação

Objetivos de Desenvolvimento Sustentável (ODS)
ODS: 16. Paz, justiça e instituições eficazes

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: https://orcid.org/0009-0001-7834-8563
- Currículo Lattes do autor: http://lattes.cnpq.br/4921374097411066

- Prof. Dr. Marcos Medeiros Raimundo
- Prof. Dr. Lehilton Lelis Chaves Pedrosa
- Prof. Dr. Guilherme Dean Pelegrina

A ata da defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria do Programa da Unidade.

Eu tive que fazer uma jogada arriscada
Tipo leão da montanha eu deixo a minha fé blindada
Eu tive que provar muito além das palavras
Depois da tempestade eu chego aonde eu não estava
(TZ da Coronel)

Acknowledgements

I would like to thank my parents, Vera Cardoso and Suemi do Nascimento, my brother Gabriel Nascimento, and my grandmother Angelina Cardoso for their unconditional support.

To my friends and colleagues from the telemarketing operation, who encouraged me to build a new path with courage – especially Richard Roque, Andrielle Moraes and Giovanna França, whose partnership was fundamental. I also thank Noah França, Nicolly França, Thamires Santos, and Murilo Augusto Santos for their kindness, encouragement, and companionship.

To the MRAI research group and to my friends from the Institute of Computing – especially Daniel Gardin, Leonardo Rezende, Caio Rhoden, and Athyrson Ribeiro – for their constant presence and valuable suggestions throughout the development of this dissertation. I also thank Gleyson Roberto do Nascimento for his attentive listening and insightful advice on academic opportunities and pathways.

To my advisor, Marcos Raimundo, for welcoming me during my career transition, for his patience, the many lessons, and all his support throughout this research.

To my co-advisors: Alessandro Chimenton, who was part of my journey in science, offering valuable corrections and insights; and Leonardo Tomazeli, for his support.

To my mentor, Kevin Murphy, for his ideas, generous perspective, and for helping me recognize my potential in science.

To my friends from UFF – Victor Julio Souza, Orlando Warlem, Evelyn Silva, Giovane Oliveira, Cristiano Campos, Baggio Castro, and so many others – for believing in me and reminding me why it's worth continuing.

To my friends and colleagues at UNICAMP, for their support during my adaptation and for their companionship throughout this journey, especially Letícia Sayuri, Filipi Chalita, Luis Gustavo Adelino, Otávio Osaki, and Lucas Kenji.

To the FEF volleyball extension group, for bringing lightness, strength, and joy amid the demands of academic life.

To UNICAMP, for the institutional support and for transforming my life – and, I hope, the lives of many around me. To the IC administrative staff, especially Priscilla Kakuzo and Wilson Bagni Jr., for making everything more feasible through their organization and warmth.

To Google DeepMind, for the financial and institutional support, and for allowing me to dream bigger.

Resumo

A Análise de Componentes Principais (PCA) e sua extensão probabilística (PPCA) são amplamente utilizadas para redução de dimensionalidade, mas não possuem garantias de *fairness*, o que pode levar a representações enviesadas. Este trabalho propõe um framework de Mistura de PPCA (MPPCA) *fair*, utilizando otimização minimax para garantir uma codificação de dados justa entre grupos sensíveis. Diferentemente de abordagens baseadas em Modelos de Mistura Gaussianos, nosso método incorpora restrições de *fairness* diretamente no processo de redução de dimensionalidade. Para isso, definimos uma função de perda que equilibra o desempenho entre os grupos sensíveis por meio de uma estratégia de otimização minimax. Fornecemos análises teóricas e validação empírica, demonstrando um melhor equilíbrio no aprendizado de representações sem comprometer as informações dos dados.

Abstract

Principal Component Analysis (PCA) and its probabilistic extension (PPCA) are widely used for dimensionality reduction but lack fairness guarantees, potentially leading to biased representations. This work introduces a fairness-aware Mixture of Probabilistic PCA (MPPCA) framework, leveraging minimax optimization to ensure equitable data encoding across sensitive groups. Unlike existing Gaussian Mixture Model-based approaches, our method integrates fairness constraints directly into the dimensionality reduction process. We achieve this by defining a log-likelihood function that balances the performance across sensitive groups using a minimax optimization stategy. We provide theoretical insights and empirical validation, demonstrating improved fairness in representation learning while preserving data information.

List of Figures

3.1 Geometric intuition of PCA — ε is the canonical basis, γ the orthonormal eigenbasis, and $\overline{x}$ the original datapoint x represented in γ. 21

5.1 Synthetic dataset sampled from a mixture of 3 Gaussians 35
5.2 Balanced setting with 4 mixture components 36
5.3 Balanced setting with 5 mixture components 36
5.4 Balanced setting with 10 mixture components 37
5.5 Unbalanced setting with 4 mixture components 37
5.6 Unbalanced setting with 5 mixture components 38
5.7 Unbalanced setting with 10 mixture components 38
5.8 Convergence of log likelihood of FR-MPPCA across groups in German Credit dataset 40
5.9 Comparison of log likelihood convergence between PCA, MPPCA and FR-MPPCA in German Credit dataset 40
5.10 Reconstruction error evolution between PCA, MPPCA and FR-MPPCA in German Credit dataset 41
5.11 ROC-AUC evolution for PCA, MPPCA and FR-MPPCA in German Credit dataset 42
5.12 Convergence of log likelihood of FR-MPPCA across groups in COMPAS dataset 43
5.13 Comparison of log likelihood convergence between PCA, MPPCA and FR-MPPCA in COMPAS dataset 44
5.14 Reconstruction error evolution between PCA, MPPCA and FR-MPPCA in COMPAS dataset 44
5.15 ROC-AUC evolution for PCA, MPPCA and FR-MPPCA in COMPAS dataset 45
5.16 Convergence of log likelihood of FR-MPPCA across groups in Adult dataset 46
5.17 Comparison of log likelihood convergence between PCA, MPPCA and FR-MPPCA in Adult dataset 47
5.18 Reconstruction error evolution between PCA, MPPCA and FR-MPPCA in Adult dataset 48
5.19 ROC-AUC evolution for PCA, MPPCA and FR-MPPCA in Adult dataset 48

Contents

1 Introduction **12**
- 1.1 Basics of Fair Machine Learning . . . 12
- 1.2 Basics of PCA . . . 14
- 1.3 Basics of PPCA . . . 15
- 1.4 A min-max approach of fair MPPCA . . . 16

2 Bibliographic Review **17**
- 2.1 Fairness in Gaussian Mixture Models and Density Estimation . . . 17
- 2.2 Fair PCA and Fair Dimensionality Reduction . . . 18
- 2.3 Fair Clustering and Probabilistic Methods . . . 18
- 2.4 Minimax Optimization for Fairness . . . 18
- 2.5 Contribution of This Work . . . 19

3 Probabilistic Principal Component Analyzers **20**
- 3.1 Principal Component Analysis . . . 20
  - 3.1.1 Geometric Intuition . . . 21
- 3.2 Probabilistic Principal Component Analysis . . . 23
- 3.3 Gaussian Mixture Models . . . 26
  - 3.3.1 Model definition . . . 26
  - 3.3.2 Maximum Likelihood Parameter Estimation . . . 26

4 Multi Group MPPCA **29**
- 4.1 EM for Weighted Mixtures of Probabilistic PCA . . . 30
- 4.2 APStar . . . 32
- 4.3 On the convergence of the EM Algorithm . . . 33

5 Experiments **35**
- 5.1 Synthetic Data Generation . . . 35
- 5.2 Convergence experiments with synthetic data . . . 36
  - 5.2.1 Balanced Data . . . 36
  - 5.2.2 Unbalanced Data . . . 37
  - 5.2.3 Convergence analysis under balanced and unbalanced settings . . . 38
- 5.3 Real datasets . . . 39
  - 5.3.1 German credit dataset . . . 39
  - 5.3.2 COMPAS dataset . . . 42
  - 5.3.3 Adult dataset . . . 46
- 5.4 Discussion . . . 48

6 Conclusion **50**

Bibliography 52

Chapter 1

Introduction

As machine learning systems become increasingly integrated into high-stakes decision-making processes—ranging from hiring and lending to healthcare and law enforcement—the issue of fairness has emerged as a critical area of concern [1]. These algorithms, while often perceived as objective, are in fact shaped by the data they are trained on and the design choices behind their implementation. This means they can perpetuate or even amplify existing societal biases, including those related to race, gender, and socioeconomic status [17]. To address these risks, the field of algorithmic fairness has focused on identifying, quantifying, and mitigating such biases throughout the machine learning pipeline [37]. One particularly important and often overlooked stage is data representation, where bias can be embedded during dimensionality reduction techniques such as Principal Component Analysis (PCA) and its probabilistic variants. This dissertation explores fairness specifically in the context of probabilistic PCA mixtures, aiming to develop representations that preserve essential information while promoting equitable treatment across different social groups.

1.1 Basics of Fair Machine Learning

Machine learning is a subfield of artificial intelligence that focuses on designing algorithms capable of learning patterns from data to make predictions or decisions. In the supervised learning setting, we are given a dataset of n training examples, each consisting of an input $x_i \in \mathbb{R}^d$ and a corresponding output label y_i. These pairs are assumed to be drawn from an unknown joint probability distribution $p_{X,Y}(x, y)$ [37]. In supervised learning, the goal is to learn a function $\hat{y} = f(x)$ that predicts the correct label y for a new, unseen input x. This function, referred to as a *model*, is learned during the training phase by minimizing some measure of prediction error on the training set. A key challenge in machine learning is generalization, or the model's ability to perform well on inputs that differ from the training samples. To aid in learning, input data are often preprocessed and transformed into a feature space in which the predictive patterns are more easily identifiable [2]. A common technique for such transformation is Principal Component Analysis (PCA), which reduces the dimensionality of the data by projecting it onto a lower-dimensional subspace that captures the most variance, thereby highlighting the structure most relevant

for pattern recognition [16].

As machine learning systems increasingly influence decisions with real-world consequences, it becomes essential to evaluate not only their predictive performance but also their societal impacts, particularly with respect to fairness. Fairness, often used interchangeably with justice, is a complex and inherently political concept that has long been studied across philosophy, law, and the social sciences. In the context of algorithmic systems, fairness connects to multiple dimensions of justice, including: distributive justice, which concerns the fairness of outcomes received by individuals or groups; procedural justice, which focuses on the fairness of the processes that lead to those outcomes; restorative justice, which aims to repair harms done; and retributive justice, which seeks to punish wrongdoing [37]. The primary goal of this work aligns with distributive justice: ensuring that algorithmic outcomes are fairly distributed across social groups. In machine learning systems, this concern is typically addressed through the lens of algorithmic fairness, particularly by measuring and correcting disparities between privileged and unprivileged groups—often defined along protected attributes such as race, gender, age, religion, or ethnicity. These attributes are not universally fixed but are identified based on legal, cultural, or ethical considerations specific to a given domain or jurisdiction [21].

Algorithmic fairness is generally discussed through two main paradigms: individual fairness and group fairness. Individual fairness asserts that individuals who are similar in relevant aspects should be treated similarly by the model. Group fairness, the primary operational lens of this work, emphasizes that groups defined by protected attributes should receive comparable average treatment or outcomes from a model [37]. There are many metrics for group fairness. One of the most classical metrics is *Statistical Parity Difference*, which directly quantifies *disparate impact*. Disparate impact occurs when a seemingly neutral policy or algorithm, though not overtly discriminatory, disproportionately disadvantages a protected group compared to a more privileged group. Statistical Parity Difference measures this by computing the difference in selection rates for the favorable outcome (e.g., loan approval, job offer) between the privileged and unprivileged groups. Specifically, it's calculated as the difference between the probability of a favorable prediction for the privileged group, $P(\hat{y}(X) = \text{fav}|Z = \text{priv})$, and the probability of a favorable prediction for the unprivileged group, $P(\hat{y}(X) = \text{fav}|Z = \text{unpr})$. A value closer to zero indicates less disparate impact and a higher degree of statistical parity [37].

$$\text{statistical parity difference} = P(\hat{y}(X) = \text{fav}|Z = \text{unpr}) - P(\hat{y}(X) = \text{fav}|Z = \text{priv}). \quad (1.1)$$

Another common fairness metric is the *Average Odds Difference*, which is based on model performance metrics. It involves two metrics in the ROC: the favorable label rate (true positive rate) and the false favorable label rate (false positive rate). This metric is computed by taking the difference between the true favorable rates of the unprivileged and privileged groups, then taking the difference of the false favorable rates between the unprivileged and privileged groups, and averaging them:

$$\begin{aligned}\text{average odds difference} = \frac{1}{2}(&P(\hat{y}(X) = \text{fav}|Y = \text{fav}, Z = \text{unpr}) \\ &- P(\hat{y}(X) = \text{fav}|Y = \text{fav}, Z = \text{priv})) \\ &+ \frac{1}{2}(P(\hat{y}(X) = \text{fav}|Y = \text{unf}, Z = \text{unpr}) \\ &- P(\hat{y}(X) = \text{fav}|Y = \text{unf}, Z = \text{priv})).\end{aligned}$$

In this metric, the true favorable rate difference and the false favorable rate difference can cancel out and hide unfairness. A way to mitigate this is to take the absolute value of these differences before averaging. A value of 0 average absolute odds difference indicates independence of $\hat{y}(X)$ and Z conditioned on Y, and this is considered a fair scenario and termed *equality of odds* [37].

Another relevant fairness metric is *Overall Accuracy Equality*. It requires similar prediction accuracy across different groups. In this case, we assume that obtaining a true negative is as desirable as obtaining a true positive [5].

While fairness is often considered at the level of predictions or decisions, it can be compromised much earlier in the machine learning pipeline—particularly during stages like data preprocessing, dimensionality reduction and representation learning. Dimensionality reduction techniques consists of finding a model $g(x)$ such that the information embedded in $y = g(x)$ is similar to the original space x. However, methods, such as PCA and its probabilistic variants, can unintentionally encode biases present in the data, leading to unfair representations that affect downstream outcomes. In dimensionality reduction, there are many ways of evaluating fairness: evaluating the fairness of the downstream classification task [15], evaluating quantity of mutual information of the latent space and the protected attributes [20], and evaluating the equality of the preserved information in the latent space [41]. Therefore, promoting group fairness in representation, guided by the principle of distributive justice, is a crucial step toward building more equitable and socially responsible machine learning systems.

1.2 Basics of PCA

Principal Component Analysis (PCA) is a widely used technique in statistical analysis and machine learning, enabling effective dimensionality reduction in domains such as data compression [27], pattern recognition [10], and image processing [23]. The goal is to find a projection matrix U that projects $x \in \mathbb{R}^d$ into latent vector $t = U(x - \mu) \in \mathbb{R}^{\underline{d}}$ ($\underline{d} < d$ and μ is the mean of x) but still preserves the information belonging to x. By projecting high-dimensional data onto a lower-dimensional subspace defined by the directions of maximum variance, PCA allows for more efficient storage, processing, and visualization of data. A key metric for evaluating the performance of PCA is the reconstruction error, which measures the loss of information when the original data is projected into a lower-dimensional space and then reconstructed back. Formally, it quantifies the difference, often via mean squared error, between the original data and its low-rank approximation.

Minimizing this error ensures that the reduced representation retains as much of the data's structure as possible, and is often the primary optimization objective in PCA.

However, this optimization is typically performed without regard to the performance of reconstruction error across different demographic groups. As a result, traditional PCA may perform well in aggregate, while disproportionately distorting the reconstructed data of certain groups: particularly those underrepresented in the training set. This becomes especially concerning in socially sensitive applications, where unequal representation fidelity can translate into downstream disparities in tasks such as classification and clustering.

Therefore, despite its effectiveness in reducing dimensionality, traditional PCA lacks mechanisms for explicitly controlling fairness across different demographic groups [28]. This limitation is especially critical in the current landscape, where biases in machine learning models can reinforce or even exacerbate social inequalities, particularly in high-impact applications [1,29,41]. Moreover, the linear nature of PCA [31] can, in some cases, contribute to disparities when the underlying data distributions differ significantly across groups.

1.3 Basics of PPCA

Probabilistic Principal Component Analysis (PPCA) extends classical PCA by framing dimensionality reduction as a density estimation problem within a generative probabilistic model [36]. Rather than solely projecting data onto a linear subspace, PPCA assumes that each observed data point is generated by first sampling a latent variable from a low-dimensional Gaussian distribution, then linearly transforming it and adding Gaussian noise. This formulation introduces a principled way to model uncertainty in both the data and its low-dimensional representation, enabling more expressive and robust statistical inferences. Understanding PPCA requires a basic notion of probabilistic modeling, particularly the idea of estimating a probability distribution from observed data. Once the distribution parameters are learned, the model can generate new data points by sampling from the latent space and projecting back into the original space. This generative perspective provides flexibility and robustness that classical PCA lacks, especially when dealing with missing data, noise, or uncertainty.

However, similar to standard PCA, PPCA does not include any mechanisms for ensuring fairness. The model's global parameter estimation can absorb and reproduce existing biases in the data, potentially leading to latent representations that reflect or amplify disparities across demographic groups. Since these representations are used in downstream tasks such as classification or clustering, any group-dependent distortions introduced by PPCA can propagate throughout the entire machine learning pipeline.

The Mixture of Probabilistic PCA (MPPCA) generalizes PPCA by allowing the data to be modeled as arising from a combination of several PPCA components. This makes MPPCA particularly effective for capturing complex, multimodal data structures that may reflect distinct subpopulations. While this expressiveness enhances modeling power, it also introduces new fairness concerns. If the mixture components align too closely with

sensitive attributes, such as gender, race, or age, they can effectively encode demographic separation within the latent space. As a result, MPPCA may unintentionally reinforce group-level disparities.

1.4 A min-max approach of fair MPPCA

This dissertation proposes a novel approach, Fair Robust MPPCA (FR-MPPCA), which integrates group fairness constraints directly into the Mixture of Probabilistic PCA (MPPCA) framework. We employ a minimax optimization strategy to optimize the reconstruction error and minimize disparity across sensitive groups. The minimax formulation suits this setting well, where conflicting goals must be addressed simultaneously. In contrast to prior work that targets fairness in Gaussian Mixture Models (GMMs) for generative tasks or clustering [26], or focuses on deterministic variants of Fair PCA [14, 19], our method directly embeds fairness into the dimensionality reduction process within a mixture-based model, an essential feature for effective representation learning in many machine learning systems.

We investigate theoretical guarantees and empirical performance to validate the effectiveness of our proposed method. The main contributions of this work are as follows:

1. A novel formulation for fair dimensionality reduction within the MPPCA framework;
2. An optimization strategy based on a minimax formulation tailored to fairness constraints;
3. A theoretical analysis providing convergence guarantees;
4. An empirical evaluation on synthetic and real world datasets, analyzing the trade-offs between fairness metrics and representation utility.

With this research, we aim to advance the intersection of fairness and representation learning, offering practical tools for mitigating bias and promoting better equity in data-driven systems.

Chapter 2

Bibliographic Review

The pursuit of fairness in machine learning models has been an active area of research, with various approaches leveraging probabilistic models, dimensionality reduction techniques, and clustering frameworks. While fairness-aware Principal Component Analysis (PCA) and Gaussian Mixture Models (GMMs) have been explored separately in multiple contexts, combining these ideas within a minimax optimization framework remains unexplored. This review highlights key contributions in the literature and their relation to our approach, which aims to incorporate fairness constraints into a mixture of probabilistic PCA (MPPCA) for dimensionality reduction. The pursuit of fairness in this work focuses primarily on notions of group fairness [38], aiming to ensure that dimensionality reduction does not introduce or amplify disparities between protected groups defined by sensitive attributes (such as gender or race).

2.1 Fairness in Gaussian Mixture Models and Density Estimation

Several works utilize Gaussian Mixture Models (GMMs) for fairness-aware data processing. For instance, [32] proposes using GMMs to estimate conditional densities for evaluating regression model bias directly at the density level, rather than transforming the problem into probability estimations. This approach provides a way to approximate fairness measures, such as Independence, Separation, and Sufficiency, by estimating relevant conditional distributions. Similarly, [26] applies GMMs to the latent space of generative diffusion models to balance facial attributes, mitigating bias without requiring retraining. This highlights the effectiveness of GMMs in capturing structural information about fairness-related subspaces. While Gaussian Mixtures (GMs) focus on modeling cluster densities and structure, dimensionality reduction techniques—such as PCA—are equally essential for preprocessing and representation learning. Like clustering, these methods also raise concerns around fairness, and will be discussed in the next section.

2.2 Fair PCA and Fair Dimensionality Reduction

Fairness-aware PCA has been studied in different settings. Some researchers propose fair PCA as an optimization problem that minimizes reconstruction error while penalizing disparities in reconstruction quality across protected groups [14]. Others define fair PCA through maximum mean discrepancy (MMD) constraints [19], ensuring that dimensionality-reduced conditional distributions remain similar across protected groups. These works demonstrate the potential of dimensionality reduction methods to encode fairness constraints while preserving helpful information in the data. However, existing fair PCA formulations do not incorporate probabilistic models or explore minimax optimization as a fairness mechanism. This serves as a motivation for adopting a minimax approach to MPPCA.

2.3 Fair Clustering and Probabilistic Methods

Clustering approaches have also been adapted to fairness constraints. Some methods propose fair soft clustering with deterministic group membership and probabilistic cluster assignments, ensuring balanced clustering solutions across groups [18]. Others introduce pairwise and community-preserving fairness notions in k-center clustering [4]. Probabilistic fair clustering techniques generalize fairness constraints by introducing uncertainty in group membership assignments [9]. These approaches indicate the relevance of probabilistic models in fairness-aware learning, though they have not been combined with probabilistic PCA for dimensionality reduction. The Mixture of Probabilistic PCA (MPPCA) [35] performs soft clustering by assigning data points probabilistically to mixture components. As such, concepts from fair clustering are directly relevant to our case study.

2.4 Minimax Optimization for Fairness

Minimax optimization has been applied in various fairness-aware learning tasks, but its use in fairness-aware dimensionality reduction is still underexplored. The theoretical advancements in related areas, such as minimax clustering and mixture models, provide a foundation that suggests our approach is feasible.

For example, [39] introduces a sparse PCA algorithm that achieves minimax-optimal convergence rates, offering a theoretical framework that could be adapted for fair dimensionality reduction. Similarly, [8] establishes fundamental limits on misclustering error, laying the groundwork for fairness in clustering, which could extend to our fairness concerns. Recent advances in dimensionality reduction extend PCA to multi-criteria optimization, addressing fairness concerns by balancing reconstruction error across groups, as seen in [33]. Minimax lower bounds are also explored in [6] for clustering in anisotropic GMMs, showing that fairness in such models is theoretically viable. Lastly, [34] demonstrates minimax optimal rates in multi-task learning, reinforcing the potential for fairness-aware models in multi-dimensional data. The minimax approach is particularly well-suited for enforcing fairness in MPPCA. Its structure aligns naturally with the trade-off between

minimizing reconstruction error and reducing disparities, effectively framing fairness as a game between accuracy and equity.

2.5 Contribution of This Work

While the existing literature has explored fairness-aware PCA, GMM-based fairness, and fair clustering separately, our approach aims to aggregate these ideas by introducing a mixture of probabilistic PCA with fairness constraints enforced via minimax optimization. By leveraging the representational power of PPCA and incorporating fairness objectives in an adversarial optimization framework, we seek to develop a principled approach that ensures fair dimensionality reduction while maintaining data fidelity.

This work stands at the intersection of multiple research directions, filling a gap in fairness-aware dimensionality reduction by integrating probabilistic models and minimax optimization techniques. Our approach builds upon existing methodologies while introducing a novel formulation that balances statistical fairness with efficient representation learning.

Chapter 3

Probabilistic Principal Component Analyzers

3.1 Principal Component Analysis

The main idea of Principal Component Analysis (PCA) is to reduce the dimensionality of a dataset consisting of a large number of correlated variables, while retaining as much as possible of the information present in the dataset. This is achieved by transforming the dataset to a new set of variables, called *principal components*, which are uncorrelated – i.e. orthogonal – and are ordered so that the first few retain the most variance present in all of the original variables. In this section, we are going to define and explore the properties of traditional PCA, using [16] as the main reference.

Suppose that $\mathbf{x}$ is a vector of d random variables, and that we are interested in the variances of the d random variables and the structure of the covariances or correlations between the d variables. It is possible to look for a small number ($\ll d$) of derived variables that condense most of the information given by these variances and correlations or covariances. The *principal component* (PC) is a linear functional of the form $\alpha_{\mathbf{1}}^{\mathbf{T}}\mathbf{x}$ (where $\alpha_1 \in \mathbb{R}^d$ is a vector of coefficients, also known as *loading coefficients*) that maximizes the variance by optimizing α_1. Expanding this expression, we have

$$\alpha_{\mathbf{1}}^{\mathbf{T}}\mathbf{x} = \alpha_{11}x_1 + \alpha_{12}x_2 + \cdots + \alpha_{1d}x_d = \sum_{j=1}^{d} \alpha_{1j}x_j. \tag{3.1}$$

After identifying the first principal component, we seek a second linear functional $\alpha_{\mathbf{2}}^{\mathbf{T}}\mathbf{x}$ that is uncorrelated with the first principal component $\alpha_{\mathbf{1}}^{\mathbf{T}}\mathbf{x}$ and has the maximum possible variance. This ensures that the second principal component captures new, independent information from the data. This process continues iteratively: at the $k-$th stage we determine a linear function $\alpha_{\mathbf{k}}^{\mathbf{T}}\mathbf{x}$ that maximizes variance while being uncorrelated with all previously computed principal components, $\alpha_{\mathbf{1}}^{\mathbf{T}}\mathbf{x}, \alpha_{\mathbf{2}}^{\mathbf{T}}\mathbf{x}, \ldots, \alpha_{\mathbf{k-1}}^{\mathbf{T}}\mathbf{x}$. In total, up to d PCs can be found. However, in practice, most of the variability in $\mathbf{x}$ is captured by a much smaller number of components, say $m \ll d$.

To determine the principal components, we consider the case where the vector of random variables $\mathbf{x}$ has a known covariance matrix Σ, whose (i, j)-th entry represents

the covariance between the i-th and j-th elements of $\mathbf{x}$. In practical scenarios where Σ is unknown, we estimate it using the sample covariance matrix $\mathbf{S}$. The k-th PC is given by $\mathbf{t}_k = \alpha_{\mathbf{k}}^{\mathbf{T}}\mathbf{x}$, where $\alpha_{\mathbf{k}}$ is an eigenvector of Σ, associated with its k-th largest eigenvalue λ_k. If we enforce the normalization constraint $\alpha_{\mathbf{k}}^{\mathbf{T}}\alpha_{\mathbf{k}} = 1$ then the variance of $\mathbf{t}_k$ is precisely λ_k.

3.1.1 Geometric Intuition

Consider that $D = \{x_1, \ldots, x_N\} \subset \mathbb{R}^d$ is a dataset measuring d features of N distinct individuals. The mean can be subtracted from all the points, so we can always assume that D has zero mean. With some reasonable assumptions, one can imagine that this cloud of points is somewhat distributed in a shape that resembles a $(d-1)$-dimensional ellipsoid $E \subset \mathbb{R}^n$, and that it contains the vast majority of points of D. The very axes we choose to collect the features' measurements can be the axis of E, but of course in general that's not the case. The statistical meaning of the orthogonal principal axes of E is: if we were to chose the axes of E as features *to begin with*, these variables would be *independent* of each other, that is, variations of just one of them don't affect the values of the others. So we have found d new *uncorrelated* features that describe the dataset D measuring the *"actual"* features of the experiment. In particular, we can think the sizes of this principal axis of E as the *relevance* of the features. The directions that stretch E more are the *principal components* of D. We describe these ideas more precisely below.

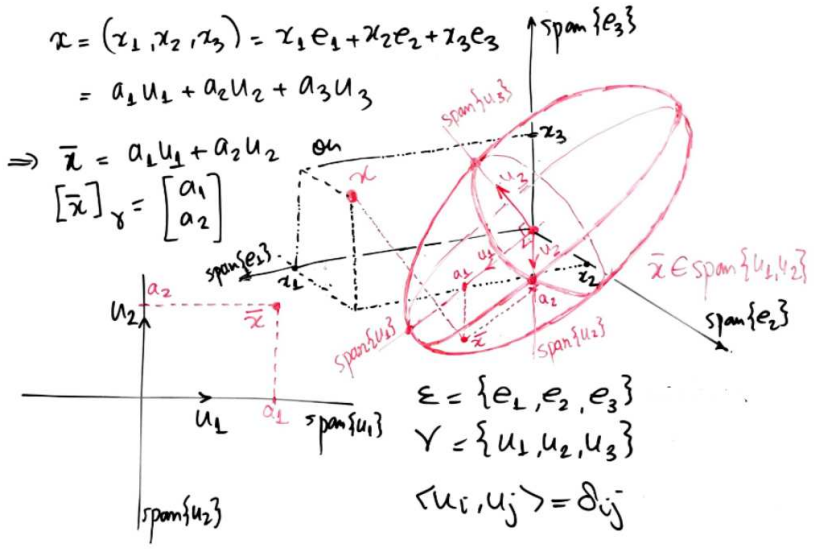

Figure 3.1: Geometric intuition of PCA — ε is the canonical basis, γ the orthonormal eigenbasis, and $\overline{x}$ the original datapoint x represented in γ.

We want to understand the "shape" of this data cloud: is it more stretched in some directions than others? How are the features related? The covariance matrix Σ of the centered data captures precisely this information about dispersion and correlation. We remember that Σ is define by

$$\Sigma = X^T X$$

where X is the matrix whose columns are the N data-vectors $x_i \in D \subset \mathbb{R}^d$. A crucial

point from linear algebra is that, since the covariance matrix is symmetric, the Spectral Theorem [12] guarantees two important facts:

- It has only real (and in this case, positive) eigenvalues.
- It has a set of eigenvectors that form an orthonormal basis for $\mathbb{R}^d$ (i.e., mutually perpendicular axes spanned by unitary vectors).

Geometrically, the covariance matrix defines the shape of the data dispersion (which is the shape of E). The link between ellipsoids and symmetric matrices is given by the quadratic forms $p : \mathbb{R}^n \longrightarrow \mathbb{R}$ suitably expressed by $p(v) = v^T Av$, where A is symmetric. If A only has positive eigenvalues, than the positive levels $p^{-1}(\{t\})$ of p are concentrical ellipsoids, and $p^{-1}(\{1\})$ has axis-sizes given precisely by the eigenvalues of A. If the data followed a multivariate Gaussian distribution, these ellipsoids would represent contours of constant probability density. More generally, they represent regions where the data has similar "statistical distance" from the center, defined by the quadratic form $p(x) = x^T\Sigma^{-1}x = \text{constant}$, known as the *Mahalanobis distance*.

We conclude that a large eigenvalue (λ_k large) means the data cloud is highly "stretched" (high variance) in the direction of the corresponding eigenvector $\mathbf{u}_k$ (a long ellipsoid axis). A small eigenvalue means the cloud is "flattened" (low variance) in that direction (a short axis of the ellipsoid).

So, what does PCA do?

1. **Finds the Axes**: It computes the eigenvectors and eigenvalues of the covariance matrix, thus identifying the principal axes of the data cloud (which are orthogonal, by the Spectral Theorem) and the variance along each axis.
2. **Ranks by Importance**: It sorts these axes (eigenvectors) according to the magnitude of their corresponding eigenvalues, from largest to smallest. The first axes in the list are called principal components. They represent the directions of greatest variance in the data—the directions in which the ellipsoid is most "stretched".
3. **Projects the data**: To reduce the dimensionality of a data x from d to q (where $q < d$), PCA selects the first q principal components (the eigenvectors with the largest eigenvalues). It then discards the remaining $d - q$ axes (those with lower variance, where the data cloud is flatter). Finally, it orthogonally projects each original data point on the $q-$ dimensional subspace spanned by the selected principal components, and gives the coordinates of this projection (relative to the previously mentioned basis of this subspace) as the *uncorrelated* feature-values of x relative to the principal components.

The result is a lower-dimensional q representation of your data that preserves as much of the original variance as possible. Geometrically, you have found the q-dimensional "view" of your data cloud that best captures its "stretching" and underlying statistical structure. Specifically, the q first coordinates of a data $x_i \in D$ relative to the basis of axis of E (ordered by decreasing magnitude of their respective eigenvalues) are the values of these q fresh discovered and "more relevant" features (Figure 3.1.) This is why the

eigenvectors of the covariance matrix are called the *principal components* of the dataset D.

3.2 Probabilistic Principal Component Analysis

In this section, we extend the discussion from standard Principal Component Analysis to its probabilistic counterpart, Probabilistic Principal Component Analysis (PPCA). We will follow the formulation proposed by [36]. This approach expands standard PCA by incorporating a probabilistic framework, modeling the data as a latent variable model with Gaussian noise.

A latent variable model seeks to relate an *observed data vector* $\mathbf{x} \in \mathbb{R}^d$ to a corresponding vector of latent variables $\mathbf{t} \in \mathbb{R}^q$, $q < d$:

$$\mathbf{x} = \mathbf{y}(\mathbf{t}; \mathbf{w}) + \epsilon, \tag{3.2}$$

where $\mathbf{y}(\cdot, \cdot)$ is a function of the latent variables $\mathbf{t}$ with parameters $\mathbf{w}$, and ϵ is an $\mathbf{x}$-independent noise process.

The most common example of a latent variable model is that of statistical *factor analysis* in which the mapping $\mathbf{y}(\mathbf{t}; \mathbf{w})$ is a linear function of $\mathbf{t}$:

$$\mathbf{x} = \mathbf{W}\mathbf{t} + \mu + \epsilon. \tag{3.3}$$

with $\mathbf{t} \sim \mathcal{N}(0, I)$, $\epsilon \sim \mathcal{N}(0, \mathbf{\Psi})$, with $\mathbf{\Psi}$ diagonal, and the $d \times q$ parameter matrix $\mathbf{W}$ contains the *factor loadings*. The observation vectors are also normally distributed $\mathbf{x} \sim \mathcal{N}(\mu, \mathbf{C})$, where the model covariance is $\mathbf{C} = \mathbf{\Psi} + \mathbf{W}\mathbf{W}^T$.

More precisely, for each linear transformation $W : \mathbb{R}^q \longrightarrow \mathbb{R}^d$ is associated to a probability distribution over $\mathbb{R}^d$, given by the conditional form:

$$p(\mathbf{x}|\mathbf{t}) = (2\pi\sigma^2)^{-d/2} \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{x} - \mathbf{W}\mathbf{t} - \boldsymbol{\mu}\|^2\right\}.$$

Our goal is to find parameters W and $\boldsymbol{\mu}$ that maximize the overall likelihood of the observed data. That is, we seek to maximize the log-likelihood function

$$\mathcal{L}(W, \boldsymbol{\mu}) = \sum_{n=1}^{N} \log p(\mathbf{x}_n),$$

where $\mathcal{L} : M_{d\times q} \times \mathbb{R}^d \to \mathbb{R}$ is the *log-likelihood*.

If the noise covariance is $\mathbf{\Psi} = \sigma^2 I$, an isotropic noise model is assumed, and the $d - q$ smallest eigenvalues of the sample covariance matrix $\mathbf{S}$ are exactly equal, then standard PCA emerges [35].

In the general Factor Analysis model, the noise term ϵ is assumed to follow a Gaussian distribution with covariance matrix $\mathbf{\Psi}$, which is typically diagonal. A particular case of this model is obtained when the noise is isotropic, i.e., $\mathbf{\Psi} = \sigma^2 I$. This leads to the Probabilistic PCA (PPCA) model [35], where the latent variable formulation remains the same, but the assumption of isotropic noise simplifies the marginal distribution of the

observed data.

For the case of isotropic noise $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$, equation 3.3 implies a probability distribution over $\mathbf{x}$-space for a given $\mathbf{t}$ of the form:

$$p(\mathbf{x}|\mathbf{t}) = (2\pi\sigma^2)^{-d/2} \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{x} - \mathbf{W}\mathbf{t} - \boldsymbol{\mu}\|^2\right\}. \tag{3.4}$$

With a Gaussian prior over the latent variables defined by

$$p(\mathbf{t}) = (2\pi)^{-q/2} \exp\left\{-\frac{1}{2}\mathbf{t}^T\mathbf{t}\right\}, \tag{3.5}$$

we obtain the marginal distribution of $\mathbf{x}$ in the form

$$p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{t})p(\mathbf{t})d\mathbf{t}, \tag{3.6}$$

$$= (2\pi)^{-d/2}|\mathbf{C}|^{-1/2} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T\mathbf{C}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right\}, \tag{3.7}$$

where the model covariance is

$$\mathbf{C} = \sigma^2\mathbf{I} + \mathbf{W}\mathbf{W}^T. \tag{3.8}$$

Having defined the marginal distribution of $\mathbf{x}$ and the covariance of the model, we can now observe that the probabilistic structure of PPCA naturally leads to a generative data model [25]. In this model, the observed data $\mathbf{x}$ are generated by a linear transformation of the latent variables $\mathbf{t}$, plus gaussian noise ϵ, i.e., $\mathbf{x} = \mathbf{W}\mathbf{t} + \epsilon$. This generative model is crucial for understanding the model's behaviour, particularly considering the marginalization over the latent variables, through the transformation matrix $\mathbf{W}$, which enables PPCA to effectively capture the structure of the data while introducing a probabilistic framework that can be used for dimensionality reduction and data generation.

Using Bayes' rule, the *posterior* distribution of the latent variables $\mathbf{t}$ given the observed $\mathbf{x}$ may be calculated:

$$p(\mathbf{t}|\mathbf{x}) = (2\pi)^{-q/2}|\sigma^{-2}\mathbf{M}|^{1/2}\times \tag{3.9}$$

$$\exp\left[-\frac{1}{2}\{\mathbf{t} - \mathbf{M}^{-1}\mathbf{W}^T(\mathbf{x} - \boldsymbol{\mu})\}^T(\sigma^{-2}\mathbf{M})\{\mathbf{t} - \mathbf{M}^{-1}\mathbf{W}^T(\mathbf{x} - \boldsymbol{\mu})\}\right], \tag{3.10}$$

where the posterior covariance matrix is given by

$$\sigma^2\mathbf{M}^{-1} = \sigma^2(\sigma^2\mathbf{I} + \mathbf{W}^T\mathbf{W})^{-1}. \tag{3.11}$$

Note that $\mathbf{M}$ is $q \times q$ while $\mathbf{C}$ is $d \times d$. The log-likehood of observing the data under this

model is

$$\mathcal{L} = \sum_{n=1}^{N} \ln\{p(\mathbf{x}_n)\} \tag{3.12}$$

$$= -\frac{N}{2}\{d\ln(2\pi) + \ln|\mathbf{C}| + \mathrm{tr}(\mathbf{C}^{-1}\mathbf{S})\}, \tag{3.13}$$

where

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \boldsymbol{\mu})(\mathbf{x}_n - \boldsymbol{\mu})^T, \tag{3.14}$$

is the sample covariance matrix of the observed $\{\mathbf{x}_n\}$. The log-likelihood is maximized when the columns of $\mathbf{W}$ span the principal subspace of the data.

We can now connect PPCA with PCA. Although both methods aim to reduce the dimensionality of the data, they do so based on different frameworks. PCA is a deterministic technique that seeks to find the subspace of lower dimension that explains the maximum variance in the data. The projection of the data is done along the principal components, which are the directions of largest variance. In contrast, PPCA introduces a probabilistic model to explain the data generation process, treating the dimensionality reduction in a stochastic manner by assuming the original data is generated by a linear transformation of latent variables with added gaussian noise. When we compute the maximum likelihood estimate (MLE) for PPCA parameters, the model converges to classical PCA as the variance of the noise tends to zero [11].

When it comes to projections, in conventional Principal Component Analysis (PCA), a data point $\mathbf{x}_n$ is mapped to its reduced-dimensionality representation $\mathbf{t}_n \in \mathbb{R}^q$ via the transformation $\mathbf{t}_n = \mathbf{U}_q^T(\mathbf{x}_n - \boldsymbol{\mu})$, and subsequently reconstructed as $\hat{\mathbf{x}}_n = \mathbf{U}_q\mathbf{t}_n + \boldsymbol{\mu}$. Here, $\boldsymbol{\mu}$ is the data mean, and $\mathbf{U}_q$ is the $d \times q$ matrix whose columns are the q principal eigenvectors (corresponding to the largest eigenvalues) of the sample covariance matrix $\mathbf{S}$.

Probabilistic PCA (PPCA) offers a related perspective but operates within a probabilistic framework. As defined earlier (Eq. 3.2), the PPCA model specifies a generative mapping from a lower-dimensional latent space (variable $\mathbf{t}_n \in \mathbb{R}^q$) to the observed data space ($\mathbf{x}_n \in \mathbb{R}^d$). To find the latent representation corresponding to an observed point $\mathbf{x}_n$ in PPCA—the conceptual analogue to PCA's projection—we invert the generative mapping $p(\mathbf{x}_n|\mathbf{t}_n)$ using Bayes' theorem. This yields the posterior distribution $p(\mathbf{z}_n|\mathbf{x}_n)$, which specifies the probability distribution over possible latent representations for $\mathbf{x}_n$ (given by Eq. 3.10).

Crucially, unlike conventional PCA which yields a single point projection $\mathbf{t}_n$, PPCA represents each data point $\mathbf{x}_n$ in the latent space via this full Gaussian posterior distribution $p(\mathbf{t}_n|\mathbf{x}_n)$. While this distribution provides a richer, probabilistic representation, a convenient point-estimate summary, analogous to the PCA projection $\mathbf{t}_n$, is given by the mean of this posterior distribution:

$$\langle \mathbf{t}_n \rangle = \mathbf{M}^{-1}\mathbf{W}_{\mathrm{ML}}^T(\mathbf{x}_n - \boldsymbol{\mu}). \tag{3.15}$$

This posterior mean $\langle \mathbf{t}_n \rangle$ represents the most likely point in the latent space corresponding to $\mathbf{x}_n$ under the PPCA model and is naturally computed during the Expectation step of the EM algorithm used to fit PPCA parameters.

3.3 Gaussian Mixture Models

To capture more information from the data and handle non-linearity more effectively, we would like to extend the concept of PPCA to a Mixture of PPCA. To achieve this, it is helpful to introduce a more general framework, Gaussian Mixture Models (GMMs).

A Gaussian Mixture Model (GMM) is a parametric probability density function represented as a weighted sum of Gaussian component densities, and its parameters are estimated from training data using the iterative Expectation-Maximization (EM) algorithm or Maximum A Posteriori (MAP) estimation from a well-trained prior model. This chapter is based on [30].

3.3.1 Model definition

A GMM is a weighted sum of M component Gaussian densities given by the equation

$$L(\mathbf{x}, \theta) = \sum_{i=1}^{M} \alpha_i p(\mathbf{x}|\boldsymbol{\mu}_i, \mathbf{C}_i) \tag{3.16}$$

where $\mathbf{x} \in \mathbb{R}^d$ is the observed data, α_i, $i = 1, \ldots, M$ are the mixture weights (or responsibilities), and $p(\mathbf{x}|\boldsymbol{\mu}_i, \mathbf{C}_i)$, $i = 1, \ldots, M$ are the component Gaussian densities. Each component density is a d-variate Gaussian function of the form,

$$p(\mathbf{x}|\boldsymbol{\mu}_i, \mathbf{C}_i) = (2\pi)^{-d/2} |\mathbf{C}_i|^{-1/2} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)^T \mathbf{C}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) \right\} \tag{3.17}$$

with mean vector $\boldsymbol{\mu}_i$, and covariance matrix $\mathbf{C}_i$. The mixture weights satisfy the constraint that $\sum_{i=1}^{M} \alpha_1 = 1$. A GMM is fully characterized by the parameters of its component distributions: the mean vectors, covariance matrices, and mixture weights. There parameters are denoted as $\theta = \{\alpha_i, \boldsymbol{\mu}_i, \mathbf{C}_i\}$, $i = 1, \ldots, M$, where α_i represents the mixture weight, $\boldsymbol{\mu}_i$ is the mean vector, and $\mathbf{C}_i$ is the covariance matrix of the i-th Gaussian component.

3.3.2 Maximum Likelihood Parameter Estimation

Given the observed data $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ where $\mathbf{x}_n \in \mathbb{R}^d$, and a GMM model parameterized by $\theta = \{\alpha_i, \boldsymbol{\mu}_i, \mathbf{C}_i\}_{i=1}^{M}$, we want to estimate the parameters θ that best match the distribution of the observed data. For this, we typically use maximum likelihood (ML) estimation. ML estimation aims to find the model parameters θ that maximize the log-likelihood of the GMM given the training data. Assuming independence between the data vectors, the log-likelihood is given by:

$$\mathcal{L}(\theta|X) = \log L(X|\theta) = \sum_{n=1}^{N} \log \left(\sum_{i=1}^{M} \alpha_i p(\mathbf{x}_n|\boldsymbol{\mu}_i, \mathbf{C}_i) \right). \tag{3.18}$$

Direct maximization of this expression concerning θ is analytically intractable due to the sum inside the logarithm. However, ML parameter estimates can be obtained iteratively using the powerful Expectation-Maximization (EM) algorithm [7].

The EM algorithm is an iterative procedure that starts with an initial guess for the parameters $\theta^{(0)}$ and alternates between two steps: the Expectation step (E-step) and the Maximization step (M-step). Each iteration t is guaranteed to yield parameters $\theta^{(t+1)}$ such that the log-likelihood does not decrease, i.e., $\mathcal{L}(\theta^{(t+1)}|X) \geq \mathcal{L}(\theta^{(t)}|X)$. The process is repeated until convergence (e.g., when the change in log-likelihood or parameters falls below a threshold).

The two steps are as follows:

1. E-Step (Expectation): In this step, we use the current parameter estimates $\theta^{(t)} = \{\alpha_i^{(t)}, \boldsymbol{\mu}_i^{(t)}, \mathbf{C}_i^{(t)}\}$ to calculate the posterior probability, or *responsibility*, $\omega_{ni}^{(t)}$ that component i was responsible for generating data point $\mathbf{x}_n$. This is computed for each data point n and each component i:

$$\omega_{ni}^{(t)} = p(z_{ni} = 1|\mathbf{x}_n, \theta^{(t)}) = \frac{\alpha_i^{(t)} p(\mathbf{x}_n|\boldsymbol{\mu}_i^{(t)}, \mathbf{C}_i^{(t)})}{\sum_{k=1}^{M} \alpha_k^{(t)} p(\mathbf{x}_n|\boldsymbol{\mu}_k^{(t)}, \mathbf{C}_k^{(t)})}. \tag{3.19}$$

Where z_{ni} is a variable that indicates if the component i was responsible for generating point $\mathbf{x}_n$. These responsibilities can be viewed as soft assignments of data points to components.

2. M-Step (Maximization): In this step, we use the responsibilities $\omega_{ni}^{(t)}$ computed in the E-step to update the parameters to $\theta^{(t+1)} = \{\alpha_i^{(t+1)}, \boldsymbol{\mu}_i^{(t+1)}, \mathbf{C}_i^{(t+1)}\}$. These updates maximize the expected complete-data log-likelihood (Q-function) [7], effectively re-estimating the parameters based on the soft assignments. The update equations are:

Mixture Weights: The new mixture weight for component i is the average responsibility it takes for the data points:

$$\alpha_i^{(t+1)} = \frac{1}{N} \sum_{n=1}^{N} \omega_{ni}^{(t)}. \tag{3.20}$$

Means: The new mean for component i is a weighted average of the data points, with weights given by the responsibilities:

$$\boldsymbol{\mu}_i^{(t+1)} = \frac{\sum_{n=1}^{N} \omega_{ni}^{(t)} \mathbf{x}_n}{\sum_{n=1}^{N} \omega_{ni}^{(t)}}. \tag{3.21}$$

Covariances: The update for the covariance matrix depends on the assumed structure. For a **full covariance matrix** $\mathbf{C}_i$, the update is:

$$\mathbf{C}_i^{(t+1)} = \frac{\sum_{n=1}^{N} \omega_{ni}^{(t)} (\mathbf{x}_n - \boldsymbol{\mu}_i^{(t+1)})(\mathbf{x}_n - \boldsymbol{\mu}_i^{(t+1)})^T}{\sum_{n=1}^{N} \omega_{ni}^{(t)}}. \tag{3.22}$$

Alternatively, if diagonal covariance matrices are assumed, $\mathbf{C}_i = \text{diag}(\sigma_{i1}^2, \ldots, \sigma_{id}^2)$, the

update for each diagonal element j is:

$$(\sigma_{ij}^2)^{(t+1)} = \frac{\sum_{n=1}^{N} \omega_{ni}^{(t)} (x_{nj} - \mu_{ij}^{(t+1)})^2}{\sum_{n=1}^{N} \omega_{ni}^{(t)}}, \tag{3.23}$$

where x_{nj} and $\mu_{ij}^{(t+1)}$ are the j-th elements of $\mathbf{x}_n$ and $\boldsymbol{\mu}_i^{(t+1)}$, respectively.

Chapter 4

Multi Group MPPCA

Inference methods based on GMMs do not inherently differentiate what we refer to as *sensitive groups*—subsets of the sample space that may be subject to bias during the data collection process. These groups often include social minorities, marginalized communities, and underrepresented demographics (e.g., ethnic minorities, women, and certain age groups). Traditional GMM-based modeling treats all points in the sample space with equal weight, overlooking potential disparities.

Our approach seeks to bridge this gap by allowing the statistical relevance of predefined sensitive groups to be explicitly adjusted through group-specific *weights*. These weights are optimized using APStar [22], a minimax-based method that we will explore in depth in later chapters.

We, first, are going to define a group-weighted loss. To do that, let $E \subset \mathbb{R}^n$ a collection of p samples with n features each. We will partition E in G subsets (sensitive groups) of the form

$$E = \cup_{g=1}^{G} E_g,$$

so that $E_g = \{x_1^g, \ldots, x_{|E_g|}^g\}$ and therefore $|E_1| + \cdots + |E_G| = p$. Let $\lambda^T = (\lambda_1, \ldots, \lambda_G) \in (\mathbb{R}^G)_+$.

In each sensitive group, the likelihood is given by $\mathcal{L}_g$. We define the vector of group-wise likelihoods as $\mathcal{L}^T = (\mathcal{L}_1, \ldots, \mathcal{L}_G)$. Consequently, the overall likelihood function of our model is expressed as a linear combination of these group likelihoods, as follows:

$$\lambda^T \mathcal{L} = \sum_k \sum_g \sum_i \lambda_{gi}\omega_{ik} \left(\ln \alpha_k - \ln \omega_{ik} - \frac{d}{2} \ln |C_k| - \frac{1}{2}(x_i - \mu_k)^T C_k^{-1}(x_i - \mu_k) \right) \tag{4.1}$$

$$= \sum_k [\ln \alpha_k \sum_g \sum_i \lambda_{gi}\omega_{ik} - \sum_g \sum_i \lambda_{gi}\omega_{ik} \ln \omega_{ik} \tag{4.2}$$

$$- \frac{d}{2} \ln |C_k| \sum_i \sum_g \omega_{ik}\lambda_{gi} - \underbrace{\frac{1}{2} \sum_g \sum_i \lambda_{gi}\omega_{ik}(x_i - \mu_k)^T C_k^{-1}(x_i - \mu_k)}_{\star})] \tag{4.3}$$

where μ_k represents the mean of each cluster, ω_{ik} denotes the responsibility of each cluster, $C_k = W_k W_k^T + \sigma_k^2 I$ is the model covariance, $\lambda_{gi} = \mathbb{I}[i = g]\lambda_g$ is the weight of each sensitive group per gaussian, α_k is the prior probability of gaussian k.

We also introduce S_k to be the sample covariance, and φ_k is introduced to simplify the notation

$$S_k = \sum_i \sum_g \lambda_{gi}\omega_{ik}(x_i - \mu_k)(x_i - \mu_k)^T$$
$$\varphi_k = \sum_i \sum_g \lambda_{gi}\omega_{ik}$$

Note that we can decompose $\lambda^T\mathcal{L}$ into losses $\mathcal{L}_k$ for each sample group $g \in \{1, \ldots, G\}$.

$$\mathcal{L}_k = \left[\varphi_k \ln \alpha_k - \sum_i \omega_{ik} \sum_g \lambda_{gi} \ln \omega_{ik} - \frac{-d\varphi_k}{2} \ln |C_k| - \frac{1}{2} \underbrace{\operatorname{tr}(C_k^{-1} S_k)}_{\star}\right] \tag{4.4}$$

Worth noticing that using the fact that $\operatorname{tr}(AB) = \sum_i \sum_j (A_{ij})(B_{ij}) = \operatorname{tr}(BA)$, we can show that

$$\operatorname{tr}(C_k^{-1}(x_i - \mu_k)\underbrace{(x_i - \mu_k)^T}) = \operatorname{tr}(\underbrace{(x_i - \mu_k)^T} C_k^{-1}(x_i - \mu_k))$$
$$\mu_k, x_i \in \mathbb{M}_{d\times 1} \Rightarrow (x_i - \mu_k)^T \in M_{1\times d}, C_k^{-1}, C_k \in M_{d\times d}$$
$$\operatorname{tr}(\underbrace{C_k^{-1}(x_i - \mu_k)}_{A}\underbrace{(x_i - \mu_k)^T}_{B}) = \operatorname{tr}(\underbrace{(x_i - \mu_k)^T}_{B}\underbrace{C_k^{-1}(x_i - \mu_k)}_{A})$$

thus, the terms marked with $\star$ in 4.3 and 4.4 are equivalent.

Now we can define the group-based max-min formulation

$$\max_g \min_{\theta\in\Theta} \mathcal{L}_g \tag{4.5}$$

The APStar [22] algorithm ensures that the weight allocation $\lambda : ||\lambda|| = 1$ maximizes the log-likelihood of the most underrepresented or statistically disadvantaged group—specifically, the group with the lowest log-likelihood. Since we can solve the weight allocation with such an algorithm, we need to calculate the partial derivatives of $\lambda^T\mathcal{L}$, which we now proceed to compute.

4.1 EM for Weighted Mixtures of Probabilistic PCA

In order to fit the proposed weighted mixture of PPCA models, we adapt the Expectation-Maximization (EM) algorithm, drawing from the formulation for standard MPPCA presented in [35, 36]. The EM algorithm iteratively refines the model parameters $\theta = \{\alpha_k, \boldsymbol{\mu}_k, \mathbf{W}_k, \sigma^2\}_{k=1}^M$ by alternating between an Expectation (E) step and a Maximization (M) step, maximizing a weighted version of the expected complete-data log-likelihood (Q-function).

Let the dataset be $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, partitioned into G sensitive groups E_g such that $g(n)$ denotes the group index for data point $\mathbf{x}_n$. Let $\lambda = (\lambda_1, \ldots, \lambda_G)$ be the vector of

fairness weights for these groups.

The objective maximized in the M-step is the weighted Q-function:

$$\mathcal{L}_{C_g} = \sum_{n=1}^{N}\sum_{i=1}^{M} z_{in} \ln\{\alpha_i p(\mathbf{x}_n, \mathbf{t}_{ni})\}. \tag{4.6}$$

where z_{ni} is the latent indicator variable (1 if point n belongs to component m, 0 otherwise), $\mathbf{t}_{ni}$ is the corresponding continuous latent variable in the q-dimensional space for component i, and $\omega_{ni}^{(t)}$ are the responsibilities calculated in the previous E-step using parameters $\theta^{(t)}$.

Starting with last iteration values for the parameters $\alpha_i, \mu_i, \mathbf{W}_i$ and σ^2 we first evaluate the posterior probabilities ω_{ni} using 3.19 and similarly evaluate the expectations $\langle \mathbf{t}_{ni} \rangle$ and $\langle \mathbf{t}_{ni}\mathbf{t}_{ni}^T \rangle$:

$$\langle \mathbf{t}_{ni} \rangle = \mathbf{M}_i^{-1}\mathbf{W}_i^T(\mathbf{x}_n - \mu_i), \tag{4.7}$$

$$\langle \mathbf{t}_{ni}\mathbf{x}_{ni}^T \rangle = \sigma^2\mathbf{M}_i^{-1} + \langle \mathbf{t}_{ni} \rangle \langle \mathbf{t}_{ni} \rangle^T, \tag{4.8}$$

with $\mathbf{M}_i = \sigma_i^2\mathbf{I} + \mathbf{W}_i^T\mathbf{W}_i$.

Then we take the expectation of $\mathcal{L}_G$ with respect to these posterior distributions to obtain

$$\langle \mathcal{L} \rangle = \sum_{g=1}^{G}\sum_{n=1}^{N}\sum_{i=1}^{K} \omega_{ni}\lambda_{ng}\{\ln \pi_i - \frac{d}{2}\ln \sigma^2 - \frac{1}{2}\mathrm{tr}(\langle \mathbf{x}_{ni}\mathbf{x}_{ni}^T \rangle) \tag{4.9}$$

$$\frac{1}{2\sigma_i^2}\|\mathbf{x}_{ni} - \mu_i\|^2 + \frac{1}{\sigma_i^2}\langle \mathbf{x}_{ni} \rangle \mathbf{W}_i^T(\mathbf{x}_n - \mu_i) \tag{4.10}$$

$$- \frac{1}{2\sigma_i^2}\mathrm{tr}(\mathbf{W}_i^T\mathbf{W}_i\langle \mathbf{t}_{ni}\mathbf{t}_{ni}^T \rangle)\}, \tag{4.11}$$

where $\langle \cdot \rangle$ denotes the expectation with respect to the posterior distributions of both $\mathbf{t}_{ni}$ and z_{ni} 3.19 and terms independent of the model parameters. The maximization of 4.11 with respect to α_i must take account of the constraint that $\sum_i \alpha_i = 1$. This can be achieved with the use of a Lagrange multiplier η and maximizing

$$\langle \mathcal{L}_G \rangle + \eta \left(\sum_{i=1}^{K} \alpha_i - 1 \right). \tag{4.12}$$

Together with the results of maximizing 4.11 with respect to the remaining parameters, this gives the following M-step equations

$$\tilde{\alpha}_i = \frac{1}{N} \sum_n \omega_{gn} \lambda_{ni} \tag{4.13}$$

$$\tilde{\mu}_i = \frac{\sum_n \omega_{ni} \lambda_{gn} (\mathbf{x}_{ni} - \tilde{\mathbf{W}}_i \langle \mathbf{t}_{ni} \rangle}{\sum_n R_{ni} \lambda_{gn}} \tag{4.14}$$

$$\tilde{\mathbf{W}}_i = \left[\sum_n \omega_{ni} \lambda_{gn} (\mathbf{x}_{ni} - \tilde{\mu}_i) \langle \mathbf{t} \rangle^T \right] \left[\sum_n \omega_{ni} \lambda_{gn} \langle \mathbf{t}_{ni} \mathbf{t}_{ni}^T \rangle \right]^{-1} \tag{4.15}$$

$$\tilde{\sigma}_i^2 = \frac{1}{d \sum_n \omega_{ni} \lambda_{gn}} \sum_n \omega_{ni} \lambda_{gn} \|\mathbf{x}_n - \tilde{\mu}_i\|^2 \tag{4.16}$$

$$= -2 \sum_n \omega_{ni} \lambda_{gn} \langle \mathbf{t}_{ni} \rangle^T \tilde{\mathbf{W}}_i^T (\mathbf{x}_n - \tilde{\mu}_i) + \sum_n R_{ni} \lambda_{gn} \mathrm{tr} \left(\langle \mathbf{t}_{ni} \mathbf{t}_{ni}^T \rangle \tilde{\mathbf{W}}_i^T \tilde{\mathbf{W}}_i \right) \tag{4.17}$$

where the symbol ˜ denotes new quantities that may be adjusted in the M-step.

4.2 APStar

Our approach iteratively refines the weight distribution λ using a minimax optimization procedure inspired by APStar [22]. Classical Gaussian Mixture Models (GMMs) estimate component weights solely based on likelihood maximization, treating all data points equally. However, this can lead to biased representations when sensitive groups are present. To address this, we introduce a fairness-aware approach that reweights the likelihood contributions of different groups using a weighting factor $\lambda = (\lambda_1, \ldots, \lambda_G)$. In this adaptation of APStar, the algorithm proceeds by first maximizing the log-likelihood $\sum \lambda_i \mathcal{L}_i$ (Line 3) and identifying the log likelihood of the worst-performing group (Line 4). During each iteration, it updates: (1) an indicator variable tracking the worst performing group's index, (2) the weighting factor λ, and (3) the weighted log-likelihood. Line 9 evaluates whether the updated λ yields improved performance for the previously worst-performing group; if satisfied, the algorithm updates $\overline{\mathcal{L}}$, the optimal parameters (θ^*, λ^*), and the best observed log-likelihood $\mathcal{L}^*$, iterating this process until convergence.

Algorithm 1 Minimax Pareto Fair Optimization

Input: parameter space: Θ, initial weights: λ, risk functions: $\mathcal{L}_a(\cdot)$, optimizer: $\arg\max_{\theta\in\Theta} \sum_{i=1}^{G} \lambda_i \mathcal{L}_i(\theta)$, $\alpha \in (0,1)$, K_{min}
Initialize:
$\theta, \mathcal{L}(\lambda) \leftarrow \arg\max_{\theta\in\Theta} \sum \lambda_i \mathcal{L}_i(\theta)$
$\overline{\mathcal{L}} \leftarrow \min_{i=1,\dots,G} \lambda_i \mathcal{L}_i(\theta)$, $K \leftarrow 1$
repeat
    $\mathbf{1}_\lambda \leftarrow \{\mathbf{1}(\lambda_i \mathcal{L}_i(\theta) \leqslant \overline{\mathcal{L}})\}_{i=1}^{G}$
    $\lambda \leftarrow \left(\alpha\lambda + \frac{1-\alpha}{K\|\mathbf{1}_\lambda\|_1^1}\mathbf{1}_\lambda\right) \frac{K}{(K-1)\alpha+1}$
    $\theta, \mathcal{L}(\lambda) \leftarrow \arg\max_{\theta\in\Theta} \sum \lambda_i \mathcal{L}_i(\theta), K \leftarrow K+1$
    if $\min_{i=1,\dots,G} \lambda_i \mathcal{L}_i(\theta) > \overline{\mathcal{L}}$ **then**
        $\overline{\mathcal{L}} \leftarrow \min_{i=1,\dots,G} \lambda_i \mathcal{L}_i(\Theta), K \leftarrow \min(K, K_{min})$
        $\theta^*, \lambda^*, \mathcal{L}^* \leftarrow \theta, \lambda, \mathcal{L}(\theta)$
    end if
until Convergence
Return: $\theta^*, \lambda^*, \mathcal{L}^*$

4.3 On the convergence of the EM Algorithm

Dempster et al. [7] define a fairly general class of EM-type algorithms and claim to prove the convergence of the method under certain assumptions. However, these assumptions were later deemed insufficient by Boyles [3], who showed that Dempster's proof contained gaps — as also noted by Wu [40], who identified a specific error between equations 3.13 and 3.14 of Theorem 2 in Dempster's original paper.

Wu proposes a technical condition on the estimated complete-data log-likelihood $Q(\varphi|\varphi')$ that ensures $|\varphi^{p+1} - \varphi^p| \longrightarrow 0$, which suggests stabilization of the sequence generated by the EM algorithm. Although this condition on Q is difficult to verify in practice, Wu argues that in the context of exponential families with suitable restrictions on model singularities, this convergence often occurs naturally, provided that the log-likelihood is upper bounded.

An important condition to guarantee that the log-likelihood is upper bounded — and consequently to ensure the existence of at least one local maximum — is to control the term $-\log|\det(WW^T + \sigma^2 I)|$, which may diverge if the covariance matrix approaches a singular matrix. In Boyles [3], a sufficient condition to avoid this issue is that the likelihood level set defined by

$$\{\mathcal{L} \geqslant \lambda_0\} = \mathcal{L}^{-1}\big([\lambda_0, +\infty)\big)$$

be a compact set, a property of any continuous and bounded by above function. However, this condition is not satisfied in the case studied by Tipping [35], whose PPCA model allows the log-likelihood to be unbounded above. In such cases, the condition $|\varphi^{p+1} - \varphi^p| \longrightarrow 0$ may also fail. To address this, Tipping directly classifies the critical points of the log-likelihood and shows that in his PPCA model — as well as in its mixture extensions — the sequence generated by EM converges to a local maximum.

In the present work, we consider an extension of this model to mixtures weighted by non-negative coefficients λ_i, each associated with one of $G \in \mathbb{N}$ prescribed different sensitive groups. Our algorithm consists of iteratively applying Tipping's EM method to the weighted log-likelihood $\lambda^T \mathcal{L}$, with G weights $\lambda_i \in \mathbb{R}$ constrained by $\sum \lambda_i = 1$ and $\lambda_i \geqslant 0$. At each iteration, after running EM with fixed λ, we identify the worst-performing group (i.e., the one with the lowest weighted log-likelihood) and update its weight, implementing a form of max-min optimization. Convergence is expected, since the update of λ does not interfere with the optimization structure of the EM algorithm, which still guarantees non-negative log-likelihood increments at each step. Moreover, the constraint that λ lies in a $(G-1)$-dimensional simplex prevents issues related to loss of compactness in the parameter space [24].

As for the issue of the log-likelihood diverging, this behavior is often caused by the term $-\log|\mathbf{C}|$, where $\mathbf{C}$ is the covariance matrix is defined as $\mathbf{C} = \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}$. This situation typically arises when the variance term σ^2 approaches zero, leading to numerical instability and an unbounded increase in the likelihood. To mitigate this problem, we introduce a regularization term by placing an Inverse-Gamma prior on σ^2.

Chapter 5

Experiments

5.1 Synthetic Data Generation

To first test the convergence of the algorithm, we created a simple dataset by sampling data from three distinct Gaussians with the following attributes:

- **Base distribution**: mixture of 3 Gaussians in $\mathbb{R}^3$
  - **Means**: $\mu_1 = (0, 0, 0)$, $\mu_2 = (3, 3, 3)$, $\mu_3 = (-3, 3, 3)$
  - **Covariances**:

$$\Sigma_1 = \begin{bmatrix} 1 & 0.1 & 0.1 \\ 0.1 & 1 & 0.1 \\ 0.1 & 0.1 & 0.1 \end{bmatrix}, \quad \Sigma_2 = \begin{bmatrix} 2 & -0.3 & -0.1 \\ -0.3 & 1 & 0.3 \\ -0.1 & 0.3 & 3 \end{bmatrix}, \quad \Sigma_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

  - **Weights**: $\lambda = (0.3, 0.4, 0.3)$
- **Sample size**: $n = 1000$

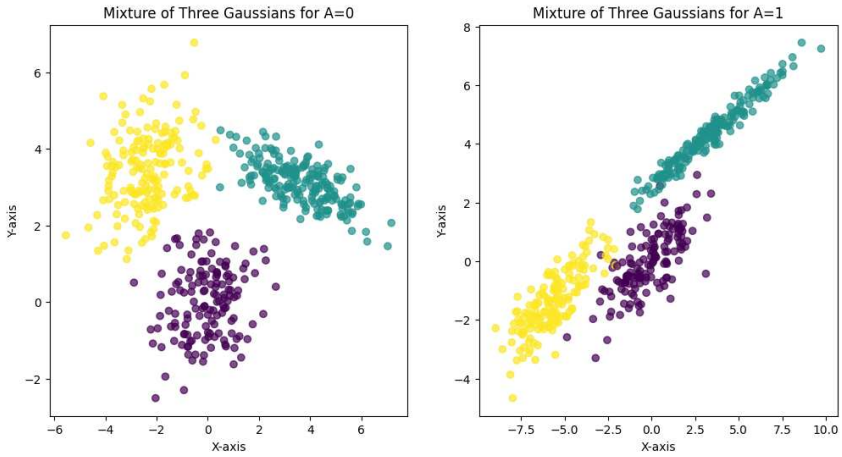

Figure 5.1: Synthetic dataset sampled from a mixture of 3 Gaussians

We evaluate the robustness of FR-MPPCA under different group distributions, particularly in the presence of class imbalance—a common scenario in real-world applications.

For the FR-MPPCA algorithm, we set the learning rate to 0.2, the maximum number of EM iterations to 100, and the maximum number of APStar iterations to 100.

5.2 Convergence experiments with synthetic data

5.2.1 Balanced Data

We first consider the case where both groups are equally represented (50% each). Labels were randomly assigned to preserve balance between groups.

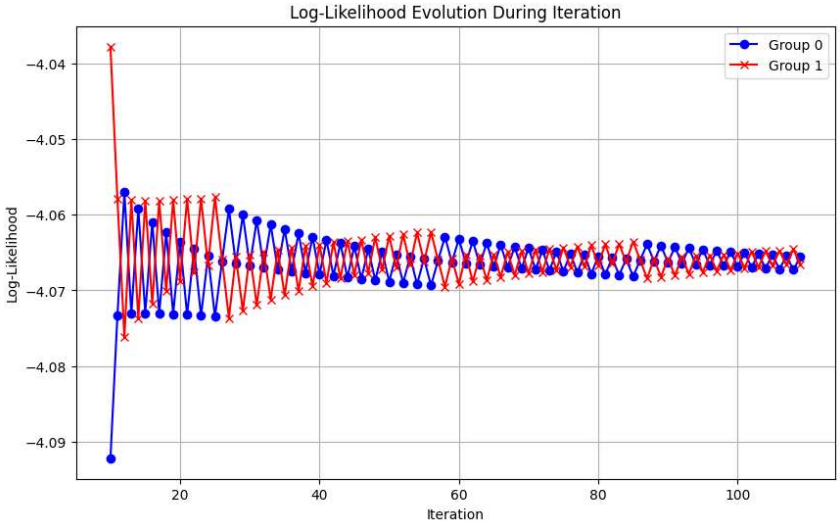

Figure 5.2: Balanced setting with 4 mixture components

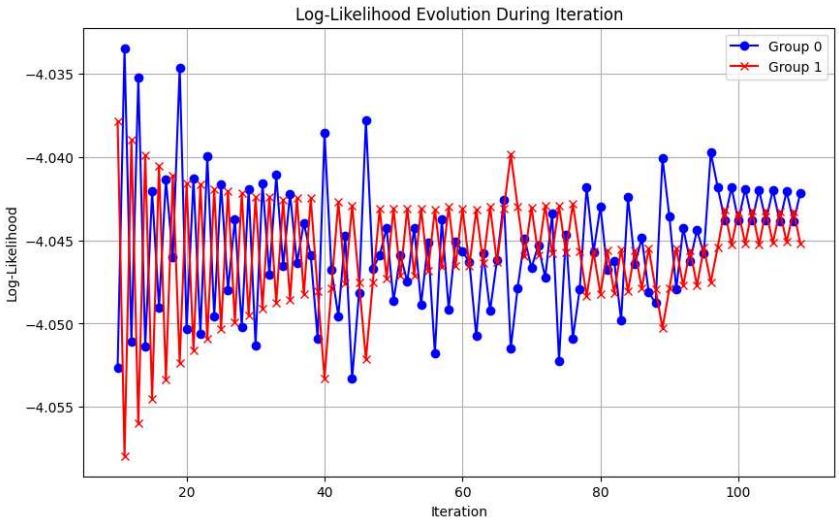

Figure 5.3: Balanced setting with 5 mixture components

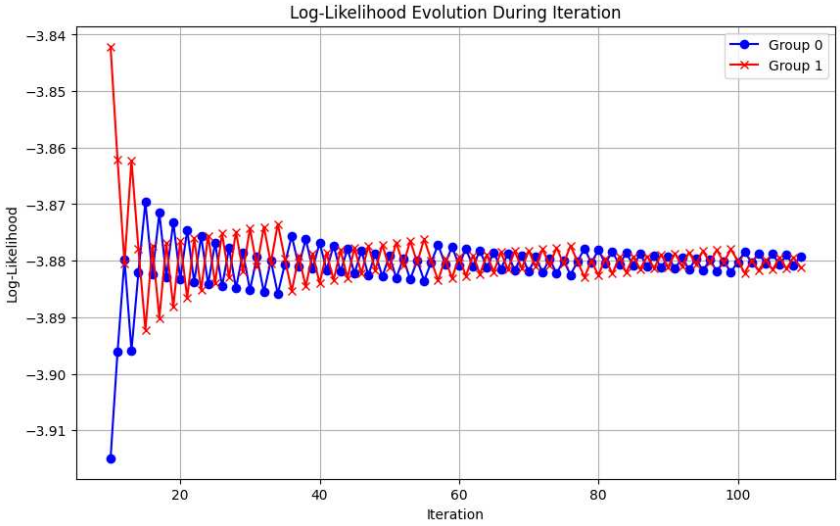

Figure 5.4: Balanced setting with 10 mixture components

5.2.2 Unbalanced Data

We then simulate a scenario where one group represents 70% of the data and the other only 30%, to assess the impact of imbalance on model performance.

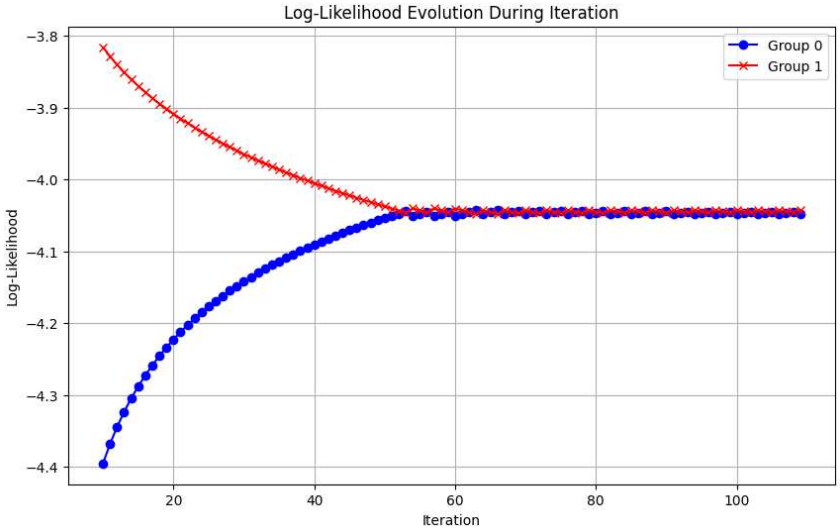

Figure 5.5: Unbalanced setting with 4 mixture components

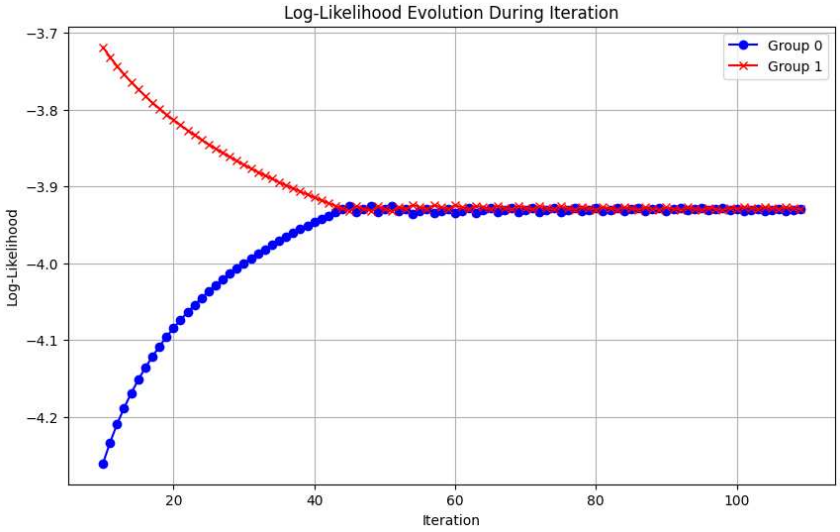

Figure 5.6: Unbalanced setting with 5 mixture components

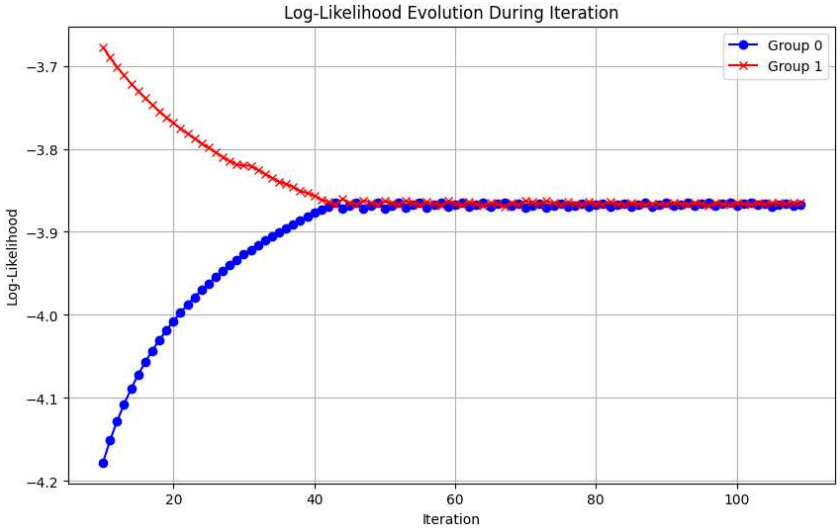

Figure 5.7: Unbalanced setting with 10 mixture components

5.2.3 Convergence analysis under balanced and unbalanced settings

We analyze the convergence behavior of FR-MPPCA in two scenarios: (i) balanced group proportions (50% for each group), and (ii) unbalanced proportions (70% for one group and 30% for the other). These experiments aim to assess the robustness of the algorithm in real-world-like situations, where imbalance is common.

Figures 5.2, 5.3, 5.4 show the log-likelihood evolution during training for different numbers of mixture components (4, 5, and 10) in the balanced case. Figures 5.5, 5.6, 5.7 present the corresponding results for the unbalanced case.

In all settings, the algorithm converges smoothly and consistently, without instability or divergence, even in the presence of class imbalance. This behavior highlights the robustness of the FR-MPPCA training procedure concerning variations in group distribution.

Although convergence speed may vary slightly depending on the degree of imbalance, the final log-likelihood values and the general convergence patterns remain similar across both scenarios.

The synthetic experiments provided controlled conditions to evaluate convergence behavior under varying group proportions. In the following sections, we move on to real-world datasets to further assess the effectiveness and fairness performance of FR-MPPCA in practical scenarios.

5.3 Real datasets

5.3.1 German credit dataset

The German Credit dataset [13] comprises 1000 entries with 20 categorical and numerical attributes. Each entry corresponds to an individual applying for credit from a bank, and each is labeled as a good or bad credit risk based on their profile. The categorical attributes include: *Status, CreditHistory, Purpose, Savings, EmploymentSince, OtherDebtors, Property, OtherInstallmentPlans, Housing, Job, Telephone*, and *ForeignWorker*. The numerical attributes are: *Duration, CreditAmount, InstallmentRate, ResidenceSince, Age, ExistingCredits, LiablePeople*, and *Target*.

We evaluate the performance of PCA, PPCA, and FR-MPPCA on this dataset, which includes sensitive group information and exhibits class imbalance (310 samples for women and 690 for men). Figure 5.8 illustrates the convergence of the log-likelihood across groups for FR-MPPCA, highlighting the effect of fairness-aware optimization. We further compare the methods in terms of log-likelihood convergence (Figure 5.9), reconstruction error over iterations (Figure 5.10), and ROC-AUC evolution (Figure 5.11).

Model Training

We fit three dimensionality reduction models on the training set (X_{train}) for comparison:

- **PPCA**: A probabilistic PCA model with 5 components is fit using the `PCA` class, serving as a baseline.
- **MPPCA**: A mixture of probabilistic PCA models is trained using the `GaussianMixture` class with 5 components. Each mixture component is modeled by a PCA with 5 latent dimensions.
- **FR-MPPCA**: Our fairness-aware extension, FR-MPPCA, is trained with 5 components and a PCA base model. It uses group labels (Z_{train}) to guide a minimax optimization procedure. We set the learning rate to 0.3, and use a randomly sampled seed for initialization. The maximum number of iterations is set to 100 for both the EM loop and the APStar outer loop.

These models are used to evaluate performance across fairness-sensitive tasks on the German Credit dataset.

Convergence of log likelihood of FR-MPPCA across groups in German Credit dataset

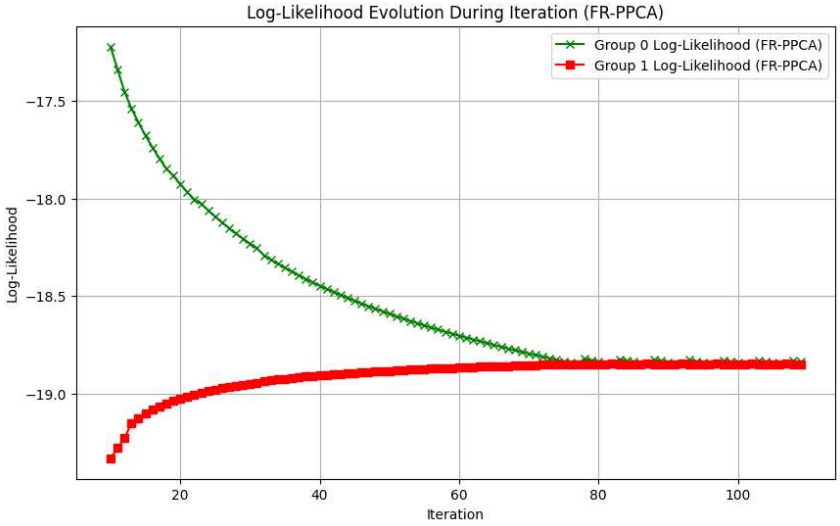

Figure 5.8: Convergence of log likelihood of FR-MPPCA across groups in German Credit dataset

In Figure 5.8, we observe that the FR-MPPCA method successfully aligns the log-likelihoods of Group 0 and Group 1, promoting convergence towards similar values. This effect becomes particularly evident after the 80^{th} iteration, highlighting the impact of the fairness-aware optimization in balancing model fit across groups.

Comparison of log likelihood convergence across methods in German Credit dataset

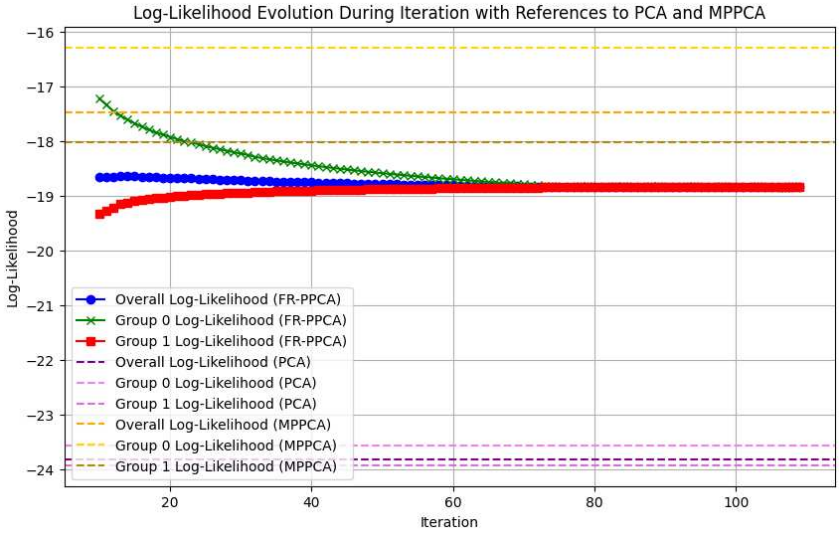

Figure 5.9: Comparison of log likelihood convergence between PCA, MPPCA and FR-MPPCA in German Credit dataset

Figure 5.9 illustrates that the overall log-likelihoods achieved by PCA and MPPCA are higher than those of FR-MPPCA, which aligns with the expected fairness-accuracy trade-off. Additionally, we observe a pronounced disparity between the log-likelihoods of Group 0 and Group 1 under PCA and MPPCA. In contrast, FR-MPPCA progressively aligns the log-likelihoods of both groups over the course of the APStar iterations, demonstrating the effect of the minimax optimization. After approximately the 76th iteration, the group-specific log-likelihoods converge toward the overall log-likelihood of FR-MPPCA, evidencing the method's ability to balance performance across sensitive groups.

Reconstruction error comparison between PCA, MPPCA and FR-MPPCA in German Credit dataset

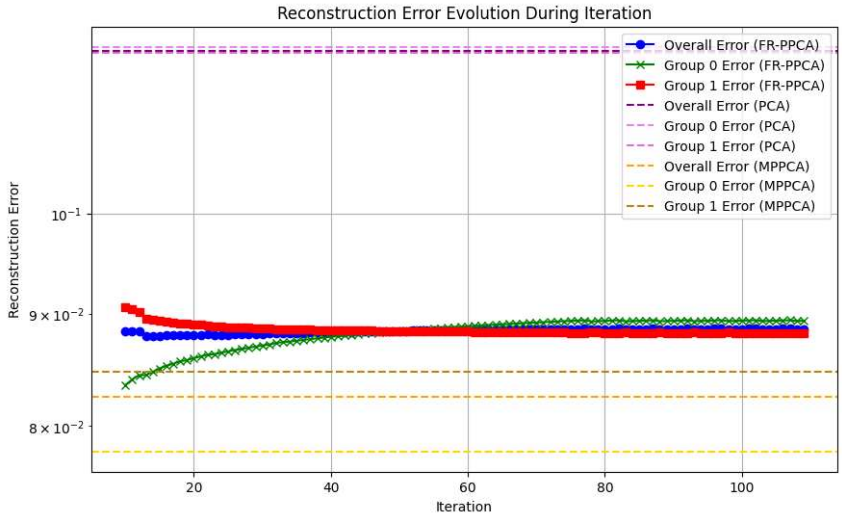

Figure 5.10: Reconstruction error evolution between PCA, MPPCA and FR-MPPCA in German Credit dataset

Figure 5.10 presents a comparison of the reconstruction error across different methods. PCA exhibits the highest overall reconstruction error, while MPPCA achieves the lowest, with FR-MPPCA positioned between the two. However, both PCA and MPPCA show a noticeable disparity in reconstruction error between groups, which can introduce bias in the learned latent representations. Despite not achieving the lowest overall error, FR-MPPCA stands out for significantly reducing this disparity, offering a more balanced reconstruction performance across groups—an essential aspect in fairness-aware modeling.

AUC Evaluation Across Methods and Groups

To evaluate the quality of the latent representations produced by PCA, MPPCA, and FR-MPPCA, we trained a logistic regression classifier with 1000 maximum iterations on the embeddings generated by each method. We used the ROC-AUC as the evaluation metric, which measures the classifier's ability to distinguish between the two classes. Importantly, we computed the AUC separately for each sensitive group defined in the dataset.

As shown in Figure 5.11, the ROC-AUC scores for PCA and MPPCA exhibit a notable disparity between groups. In contrast, FR-MPPCA shows evolving ROC-AUC values

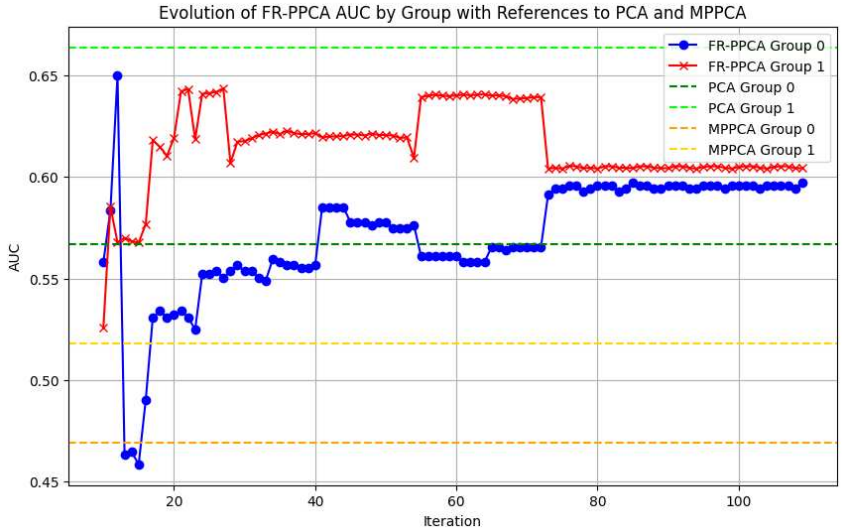

Figure 5.11: ROC-AUC evolution for PCA, MPPCA and FR-MPPCA in German Credit dataset

throughout training, reflecting the influence of the minimax optimization. Notably, after the 80th iteration, the AUC scores for both groups converge to similar values, indicating a significant reduction in performance disparity.

5.3.2 COMPAS dataset

The COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) dataset is one of the most widely used benchmarks in algorithmic fairness research. It contains information about individuals assessed for their risk of recidivism and has gained attention due to concerns about racial and gender bias in automated risk assessment tools used in the U.S. criminal justice system.

For our analysis, we use a filtered version of the dataset containing 6,172 samples and 9 attributes. These attributes include both demographic and criminal history features. The numerical features are: *age, number of juvenile felony charges, number of juvenile misdemeanors, other juvenile offenses,* and *total number of prior charges.* The categorical features are: *sex, current charge degree,* and *race.* In our experiments, race is used as the sensitive attribute for fairness evaluation, given the dataset's imbalance: it contains 3,175 samples of African-American individuals and 2,103 of white individuals.

The COMPAS dataset is particularly relevant for testing fairness-aware dimensionality reduction techniques, as it reflects well-documented group disparities. We use this dataset to compare the performance of PCA, PPCA, and FR-MPPCA in terms of both reconstruction quality and fairness of representations across racial groups.

Model Training

To evaluate the fairness and predictive performance of each dimensionality reduction method, we trained three models: PCA, MPPCA, and our proposed FR-MPPCA, each configured with five latent components.

After the dimensionality reduction step, we used a logistic regression classifier with a maximum of 1000 iterations to assess the predictive power of the learned latent representations. The classifier was trained on the transformed training set and evaluated on the test set using the ROC-AUC metric.

FR-MPPCA was configured with a maximum of 100 iterations for both the Expectation-Maximization (EM) algorithm and the outer loop of the APStar procedure. As it follows a minimax optimization strategy, we monitored the ROC-AUC score at each iteration of the APStar loop. This iterative tracking allowed us to analyze in detail how fairness—measured in terms of performance parity between groups—evolves throughout the training process. The ROC-AUC values computed at each step were systematically recorded and later organized into a structured table to facilitate comparative analysis among the methods.

Convergence of log likelihood of FR-MPPCA across groups in COMPAS dataset

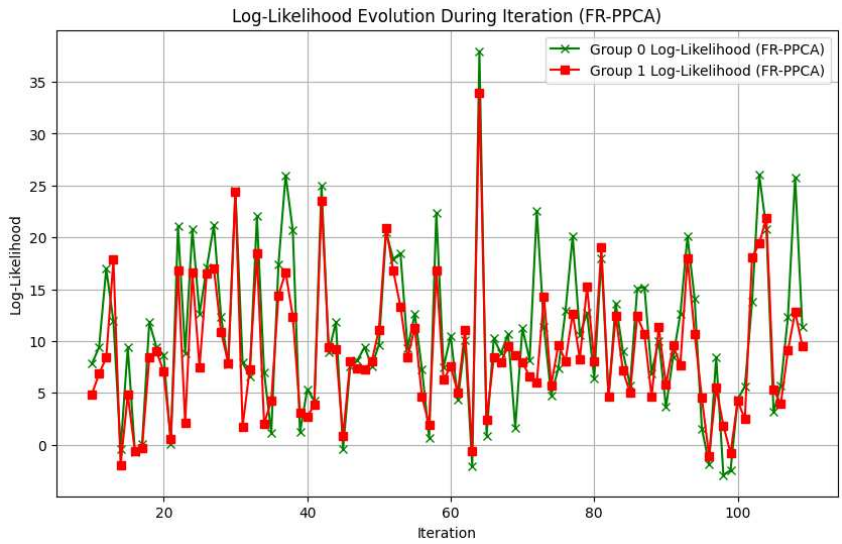

Figure 5.12: Convergence of log likelihood of FR-MPPCA across groups in COMPAS dataset

We observe in Figure 5.12 a convergence of the log-likelihood values for Group 1 (African American individuals) and Group 0 (White individuals) toward a common value. This behavior reflects the effect of the minimax optimization approach used in FR-MPPCA, which seeks to maximize the minimum group-level log likelihood. The log-likelihood exhibits considerable variability across iterations, indicating that this dataset is more complex than the German Credit dataset. Over time, we observe higher log-likelihood values for both Group 0 and Group 1 compared to the inital iterations, along with a reduction in the disparity between the groups compared to the initial iterations. In Figure 5.13, we observe that FR-MPPCA achieves the second highest overall log-likelihood among the three methods,with PCA showing the weakest performance both overall and within each group. Notably, FR-MPPCA also demonstrates the lowest disparity between

groups, effectively balancing accuracy and fairness. This highlights the strength of the minimax-based optimization in reducing group-level performance gaps without severely compromising the overall log-likelihood.

Comparison of log likelihood convergence across methods in COMPAS dataset

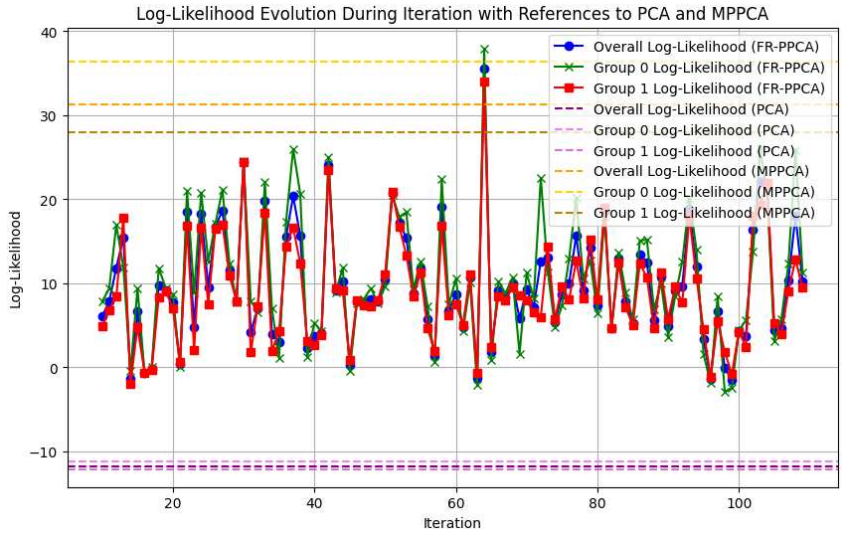

Figure 5.13: Comparison of log likelihood convergence between PCA, MPPCA and FR-MPPCA in COMPAS dataset

Reconstruction error comparison between PCA, MPPCA and FR-MPPCA in COMPAS dataset

Figure 5.14: Reconstruction error evolution between PCA, MPPCA and FR-MPPCA in COMPAS dataset

In Figure 5.14, PCA exhibits the highest overall reconstruction error, followed by FR-MPPCA, while MPPCA achieves the best performance in terms of overall reconstruction

accuracy, slightly better than the overall performance of FR-MPPCA. However, when analyzing the group-wise performance under FR-MPPCA, we observe a consistent divergence in the reconstruction errors of Group 0 and Group 1 over the iterations. This behavior reflects the dynamics of the minimax optimization, which seeks to decrease the log-likelehood of the worst groups but it is not necessarily related to the minimization of the group with worst reconstruction error (Group 0). This suggests a greater intrinsic variance or heterogeneity within this group, which may make it harder to model using low-dimensional representations. Consequently, the observed divergence in reconstruction error highlights not only the limitations of existing modeling approaches but also the complexity of the fairness-accuracy tradeoff in real-world datasets like COMPAS. Achieving parity in reconstruction quality, therefore, may demand more expressive models, additional regularization strategies, or alternative representations that better capture the nuances within each group.

AUC Evaluation Across Methods and Groups

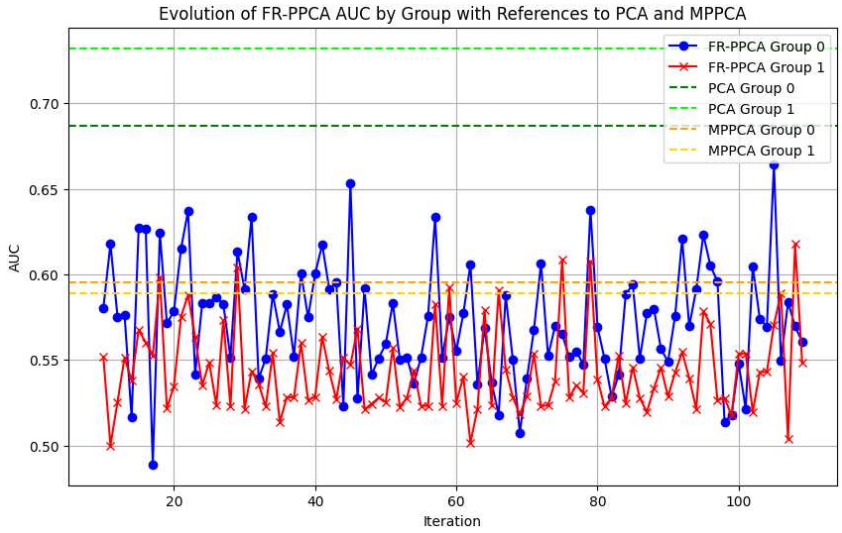

Figure 5.15: ROC-AUC evolution for PCA, MPPCA and FR-MPPCA in COMPAS dataset

The ROC AUC for PCA is approximately 0.74 for Group 1 and 0.68 for Group 0, representing the best overall performance in terms of classification accuracy. Among the probabilistic models, MPPCA achieves the second-best overall performance, with a more stable classification accuracy across groups. FR-MPPCA, by contrast, struggles with this more complex dataset, resulting in a spiky ROC AUC trajectory across iterations. While FR-MPPCA eventually reaches a relatively high ROC AUC for the underrepresented group (Group 0), its performance fluctuates more significantly than MPPCA's, reflecting the challenge of balancing group-wise fairness and predictive reliability in a complex data landscape.

5.3.3 Adult dataset

In this section, we evaluate the models on the Adult Income dataset, a benchmark for fairness studies. The dataset contains 32561 instances, each with demographic and socioeconomic attributes. We focus on the sensitive attribute sex, encoded as 0 for men and 1 for women. The dataset is imbalanced, with women representing about one-third of the samples.

The categorical variables are: *workclass, education, marital status, occupation, relationship, race, sex,* and *native country.* The numerical variables are: *age, education num, capital gain, capital loss,* and *hours per week.* We then split the data into training and test sets and apply PCA, MPPCA, and FR-MPPCA to assess both reconstruction quality and fairness across groups.

Model training

To evaluate the effect of dimensionality reduction on both fairness and predictive accuracy, we apply three methods to the preprocessed training data. First, PCA, a standard linear reduction technique; second, MPPCA, a mixture of multiple PCA components modeled via Gaussian mixtures; and third, FR-MPPCA, our fairness-aware extension that minimizes the worst-case log-likelihood across sensitive groups (sex) during training. All methods reduce the data to five components. For FR-PPCA, the maximum number of EM steps was set to 100, and the APStar optimization was also limited to 100 steps. A logistic regression classifier is then trained on the reduced representations, with 1000 maximum iterations, and ROC-AUC is computed on the test set, both overall and separately for men and women. For FR-MPPCA, we also track AUC across iterations to observe the fairness–performance trade-off. Final results highlight how each method balances predictive power and group fairness.

Convergence of log likelihood of FR-MPPCA across groups in Adult dataset

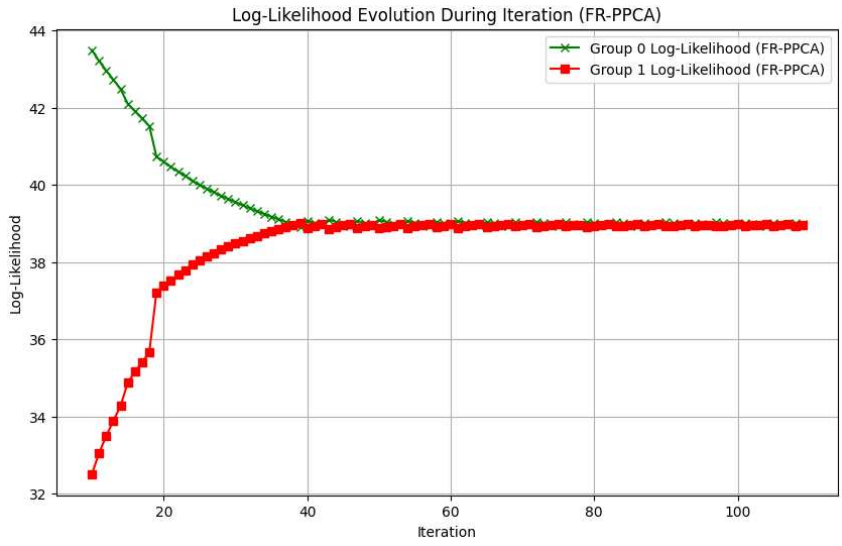

Figure 5.16: Convergence of log likelihood of FR-MPPCA across groups in Adult dataset

We observe in Figure 5.16 that after approximately the 40th iteration of APStar, the log-likelihood values for both groups begin to converge, demonstrating that FR-MPPCA effectively reduces group disparities—a key goal of our fairness-aware approach. Initially, Group 0 (men) exhibited a higher log-likelihood than Group 1; however, as FR-MPPCA progresses, the log-likelihood for Group 0 decreases, thereby favoring the performance of Group 1, which is achieved through the minimax APStar strategy.

Comparison of log likelihood convergence across methods in Adult dataset

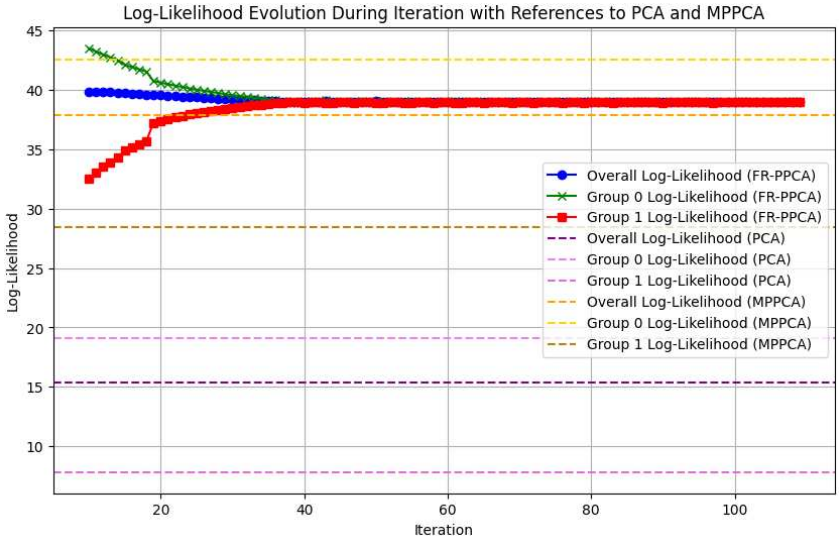

Figure 5.17: Comparison of log likelihood convergence between PCA, MPPCA and FR-MPPCA in Adult dataset

In this experiment, we observe in Figure 5.17 PCA shows the lowest overall log-likelihood and exhibits a considerable disparity between Group 0 and Group 1. MPPCA achieves better overall performance but still presents a notable gap between groups, with Group 0 reaching the highest log-likelihood among all methods. In contrast, our proposed method, FR-MPPCA, achieves the highest overall log-likelihood while substantially reducing the disparity between groups, effectively balancing fairness and performance.

Reconstruction error comparison between PCA, MPPCA and FR-MPPCA in Adult dataset

In Figure 5.18 FR-MPPCA delivers the best performance in terms of both overall and Group 1 reconstruction error. After around the 40th iteration, the reconstruction errors for both groups converge, highlighting its effectiveness in improving fairness. PCA shows the worst performance in overall reconstruction error and exhibits significant disparity between the groups. MPPCA ranks second in performance, though it still displays a high disparity between groups.

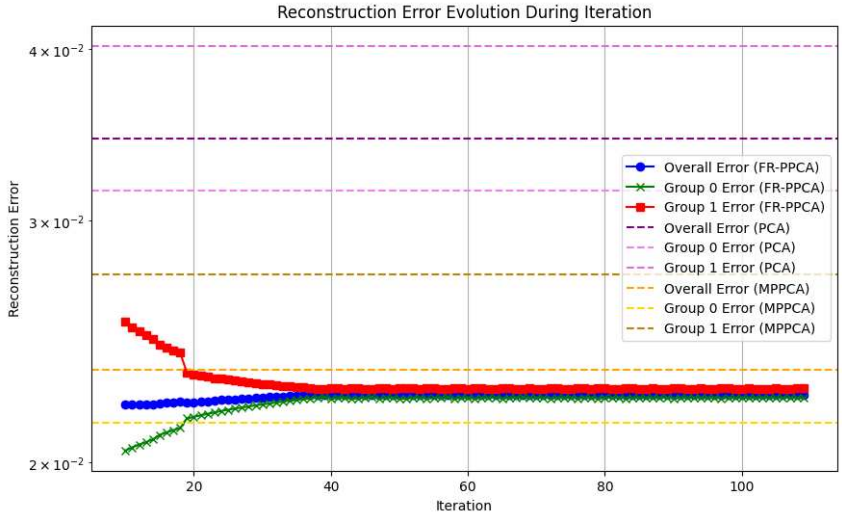

Figure 5.18: Reconstruction error evolution between PCA, MPPCA and FR-MPPCA in Adult dataset

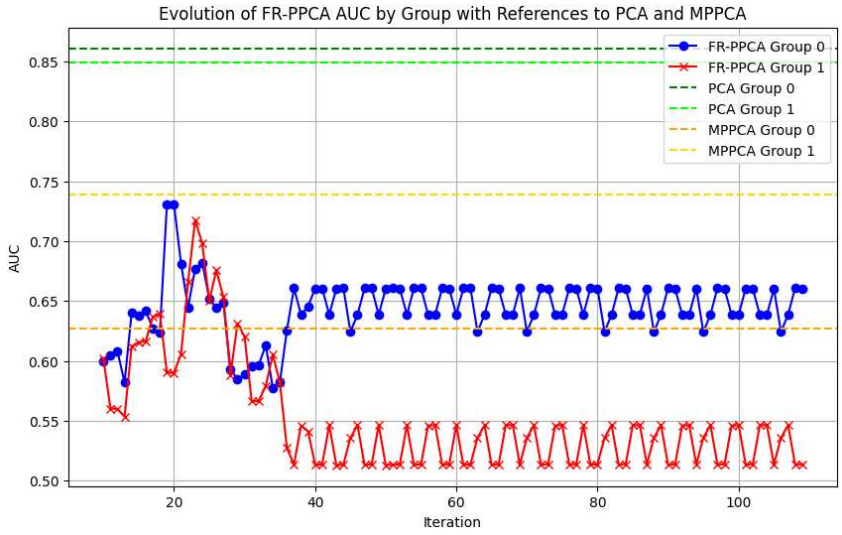

Figure 5.19: ROC-AUC evolution for PCA, MPPCA and FR-MPPCA in Adult dataset

AUC Evaluation Across Methods and Groups

We can observe in Figure 5.19 that PCA achieves the best overall ROC AUC performance, but still exhibits significant group disparity. In the case of MPPCA, the ROC AUC for group 1 (the underrepresented group) is higher than that of group 0, indicating better performance for the less represented group. For FR-MPPCA, we observe an initial drop in performance; however, after approximately the 35th iteration, the ROC AUC for group 0 (the overrepresented group) surpasses that of group 1.

5.4 Discussion

The synthetic and real-world-based experiments demonstrate the algorithm's ability to find representations that achieve a log-likelihood comparable to that of all groups. De-

spite this, real-world experiments showed that reducing the reconstruction error does not necessarily result in better performance in classification tasks. Our experiments demonstrated that PPCA is generally superior to mixtures of PPCAs, including the proposed methodology. Still, FR-MPPCA usually achieves a fairer performance across groups than MPPCA, in some of its iterations.

We believe that further investigation is needed into the nature of MPPCA's reconstruction and data distribution to understand the performance for every case.

Chapter 6

Conclusion

Dimensionality reduction is a crucial step in many machine learning pipelines, but standard methods like PCA and PPCA can inadvertently amplify existing biases in data, resulting in unfair latent representations for different demographic groups. Seeking to mitigate this issue, this dissertation introduced FR-MPPCA, a novel probabilistic dimensionality reduction technique based on Mixtures of Probabilistic PCA (MPPCA). FR-MPPCA was specifically designed to minimize reconstruction error disparities across predefined sensitive groups by incorporating group-specific weights λ_g into the learning objective, optimized via a minimax strategy (APStar). The method's performance was empirically evaluated against traditional PCA and MPPCA.

Our empirical results demonstrate the efficacy of FR-MPPCA. The proposed model significantly reduced reconstruction error disparities compared to baseline methods, while maintaining competitive overall reconstruction quality. The underlying minimax optimization framework successfully balanced the competing objectives of data fidelity and inter-group fairness, even in scenarios with class imbalance. This confirms the feasibility of achieving fairer low-dimensional representations without substantial sacrifices in model utility.

Despite promising results, this study has limitations. Our current empirical validation primarily focused on binary sensitive attributes; future research should extend the evaluation to scenarios involving multiple, potentially intersecting, groups and investigate performance on more complex, high-dimensional real-world datasets where fairness considerations are often critical. Additionally, a deeper theoretical analysis of the geometric properties of the fairness-constrained parameter space could reveal insights into the solution structure and potentially guide the development of more sophisticated optimization techniques beyond the current EM/APStar approach.

Future work could explore the application of FR-MPPCA in diverse domains facing fairness challenges, such as processing image, biomedical, or large-scale data. Its potential as a fairness-aware feature compression technique warrants investigation. Furthermore, adapting FR-MPPCA or its principles to learn fair latent representations within large language models (LLMs) or other complex generative architectures presents a promising avenue for promoting equity in cutting-edge AI systems.

In conclusion, FR-MPPCA offers a principled and effective method for integrating group fairness directly into probabilistic dimensionality reduction. By uniquely combin-

ing the representational flexibility of Mixture Models (MPPCA) with a robust minimax optimization strategy for fairness, this work addresses a critical gap in the development of equitable machine learning pipelines. It represents a tangible step towards building data analysis tools that are not only accurate and efficient but also demonstrably fairer in their treatment of diverse populations, contributing to the broader goal of responsible and ethical AI.

Bibliography

[1] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning: Limitations and Opportunities*. MIT Press, 2023.

[2] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[3] Russell A Boyles. On the convergence of the em algorithm. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 45(1):47–50, 1983.

[4] Brian Brubach, Darshan Chakrabarti, John Dickerson, Samir Khuller, Aravind Srinivasan, and Leonidas Tsepenekas. A pairwise fair and community-preserving approach to k-center clustering. In *International conference on machine learning*, pages 1178–1189. PMLR, 2020.

[5] Alycia N Carey and Xintao Wu. The statistical fairness field guide: perspectives from social and formal sciences. *AI and Ethics*, 3(1):1–23, 2023.

[6] Xin Chen and Anderson Y Zhang. Optimal clustering in anisotropic gaussian mixture models. *arXiv preprint arXiv:2101.05402*, 2021.

[7] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1):1–22, 1977.

[8] Maximilien Dreveton, Alperen Gözeten, Matthias Grossglauser, and Patrick Thiran. Universal lower bounds and optimal rates: Achieving minimax clustering error in sub-exponential mixture models. *arXiv preprint arXiv:2402.15432*, 2024.

[9] Seyed Esmaeili, Brian Brubach, Leonidas Tsepenekas, and John Dickerson. Probabilistic fair clustering. *Advances in Neural Information Processing Systems*, 33:12743–12755, 2020.

[10] A Ferraz, E Esposito, RE Bruns, and N Durán. The use of principal component analysis (pca) for pattern recognition in eucalyptus grandis wood biodegradation experiments. *World Journal of Microbiology and Biotechnology*, 14:487–490, 1998.

[11] Benyamin Ghojogh, Ali Ghodsi, Fakhri Karray, and Mark Crowley. Factor analysis, probabilistic principal component analysis, variational inference, and variational autoencoder: Tutorial and survey. *arXiv preprint arXiv:2101.00734*, 2021.

[12] Paul R Halmos. What does the spectral theorem say? *The American Mathematical Monthly*, 70(3):241–247, 1963.

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

[14] Fei Huang, Junhao Shen, Yanrong Yang, and Ran Zhao. Fairness-aware principal component analysis for mortality forecasting and annuity pricing. *arXiv preprint arXiv:2412.04663*, 2024.

[15] Taeuk Jang and Xiaoqian Wang. Fades: Fair disentanglement with sensitive relevance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12067–12076, 2024.

[16] Ian Jolliffe. Principal component analysis. *Encyclopedia of statistics in behavioral science*, 2005.

[17] Maximilian Kasy. Algorithmic bias and racial inequality: a critical review. *Oxford Review of Economic Policy*, 40(3):530–546, 2024.

[18] Rune D Kjærsgaard, Pekka Parviainen, Saket Saurabh, Madhumita Kundu, and Line KH Clemmensen. Fair soft clustering. In *27th International Conference on Artificial Intelligence and Statistics*, pages 1270–1278. Proceedings of Machine Learning Research, 2024.

[19] Junghyun Lee, Gwangsu Kim, Mahbod Olfat, Mark Hasegawa-Johnson, and Chang D Yoo. Fast and efficient mmd-based fair pca via optimization over stiefel manifold. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 7363–7371, 2022.

[20] Ji Liu, Zenan Li, Yuan Yao, Feng Xu, Xiaoxing Ma, Miao Xu, and Hanghang Tong. Fair representation learning: An alternative to mutual information. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1088–1097, 2022.

[21] Renqiang Luo, Tao Tang, Feng Xia, Jiaying Liu, Chengpei Xu, Leo Yu Zhang, Wei Xiang, and Chengqi Zhang. Algorithmic fairness: A tolerance perspective. *arXiv preprint arXiv:2405.09543*, 2024.

[22] Natalia Martinez, Martin Bertran, and Guillermo Sapiro. Minimax pareto fairness: A multi objective perspective. In *International conference on machine learning*, pages 6755–6764. PMLR, 2020.

[23] M Mudrova and Aleš Procházka. Principal component analysis in image processing. In *Proceedings of the MATLAB technical computing conference, Prague*, 2005.

[24] James Munkres. Topology james munkres second edition.

[25] Kevin P Murphy. *Probabilistic machine learning: Advanced topics*. MIT press, 2023.

[26] Basudha Pal, Arunkumar Kannan, Ram Prabhakar Kathirvel, Alice J O'Toole, and Rama Chellappa. Gaussian harmony: Attaining fairness in diffusion-based face generation models. *arXiv preprint arXiv:2312.14976*, 2023.

[27] Jyoti Pareek and Joel Jacob. Data compression and visualization using pca and t-sne. In *Advances in Information Communication Technology and Computing: Proceedings of AICTC 2019*, pages 327–337. Springer, 2021.

[28] Guilherme Dean Pelegrina and Leonardo Tomazeli Duarte. A novel approach for fair principal component analysis based on eigendecomposition. *IEEE Transactions on Artificial Intelligence*, 2023.

[29] Shaina Raza. Connecting fairness in machine learning with public health equity. In *2023 IEEE 11th International Conference on Healthcare Informatics (ICHI)*, pages 704–708. IEEE, 2023.

[30] Douglas A Reynolds et al. Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663), 2009.

[31] Juan Luis Suárez, Salvador García, and Francisco Herrera. A tutorial on distance metric learning: Mathematical foundations, algorithms, experimental analysis, prospects and challenges. *Neurocomputing*, 425:300–322, 2021.

[32] Wei Sun, Xuning Tang, and Kuo-Chu Chang. Regression model bias evaluation by estimating conditional densities with gaussian mixtures. In *2024 27th International Conference on Information Fusion (FUSION)*, pages 1–10. IEEE, 2024.

[33] Uthaipon Tantipongpipat, Samira Samadi, Mohit Singh, Jamie H Morgenstern, and Santosh Vempala. Multi-criteria dimensionality reduction with applications to fairness. *Advances in neural information processing systems*, 32, 2019.

[34] Ye Tian, Haolei Weng, and Yang Feng. Unsupervised multi-task and transfer learning on gaussian mixture models. *arXiv preprint arXiv:2209.15224*, 2022.

[35] Michael E Tipping. Mixtures of probabilistic principal component analysers. *Neural computation*, 11:435–474, 1998.

[36] Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 61(3):611–622, 1999.

[37] Kush R Vashney. *Trustworthy machine learning*. Independently published, 2022.

[38] Sahil Verma and Julia Rubin. Fairness definitions explained. In *Proceedings of the international workshop on software fairness*, pages 1–7, 2018.

[39] Zhaoran Wang, Huanran Lu, and Han Liu. Tighten after relax: Minimax-optimal sparse pca in polynomial time. *Advances in neural information processing systems*, 27, 2014.

[40] CF Jeff Wu. On the convergence properties of the em algorithm. *The Annals of statistics*, pages 95–103, 1983.

[41] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International conference on machine learning*, pages 325–333. PMLR, 2013.