FAIRSPLIT: Decomposing the Embedding Space for Fair Classification

Andrii Shkabrii[1,2]* **Anna Beer**[3,1] **Ylli Sadikaj**[1] **Claudia Plant**[1,4]
[1]Faculty of Computer Science, University of Vienna, Vienna, Austria
[2]Doctoral School Computer Science, University of Vienna, Vienna, Austria
[3]Webster Vienna Private University, Vienna, Austria
[4]Research Network Data Science, University of Vienna, Vienna, Austria

Abstract

Fair classification faces a twofold challenge: maintaining high classification accuracy while limiting reliance on sensitive information. Our method FAIRSPLIT achieves this by learning a decomposition of the embedding space into a Fair Space used for prediction and a Sensitive Space that isolates sensitive information. During training, the model must simultaneously learn the classification task and separate fair from sensitive information. The Sensitive Space gives sensitive attributes a destination, so the Fair Space can be used for prediction without relying on them. This decomposition is guided via a learnable mask and the Hilbert-Schmidt Independence Criterion (HSIC) as a measure of statistical dependence. We provide theoretical guarantees showing that the demographic parity gap is upper-bounded by both (i) the learned dimensionality of the Fair Space and (ii) the HSIC between the Fair Space and sensitive attributes. FAIRSPLIT effectively handles multiclass tasks and non-binary sensitive attributes, overcoming key limitations of existing fair classification methods. Experiments on established fairness benchmarks show that FAIRSPLIT achieves competitive accuracy-fairness trade-offs. Codebase

1 Introduction

With the expanding use of machine learning (ML) models in high-stakes domains, including hiring [47], medicine [15], criminal justice [2], credit scoring [23], and face recognition [51, 18, 7, 42], numerous studies have shown that these systems can perpetuate historical biases affecting underrepresented demographic groups. Consequently, group fairness has gained attention in ML tasks like classification [15, 36]. However, state-of-the-art methods for fair classification remain limited: many methods are designed for binary classification or assume binary sensitive attributes, restricting their applicability to realistic multiclass settings [28, 21, 37, 8]. While empirical improvements in fairness metrics are frequently reported, theoretical guarantees are scarce. Most in-processing methods for fair classification regularize dependence between predictions and sensitive attributes during training. However, acting at the prediction level provides limited guarantees for the resulting classifier and does not directly shape the learned representation it operates on.

We introduce FAIRSPLIT, an in-processing method for fair classification that acts on the representation instead: it learns which embedding dimensions to use for prediction and which to exclude because they carry sensitive information (Figure 1). Specifically, our method splits the embedding space into (i) a Fair Space with minimized dependence on the sensitive attribute using the Hilbert-Schmidt Independence Criterion (HSIC) [20], and (ii) a complementary Sensitive Space that captures residual sensitive information and is excluded from the classifier. The dimensionality of the Fair Space is itself learned during training rather than fixed in advance. We prove that a continuous relaxation of the

*Corresponding author: andrii.shkabrii@univie.ac.at

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

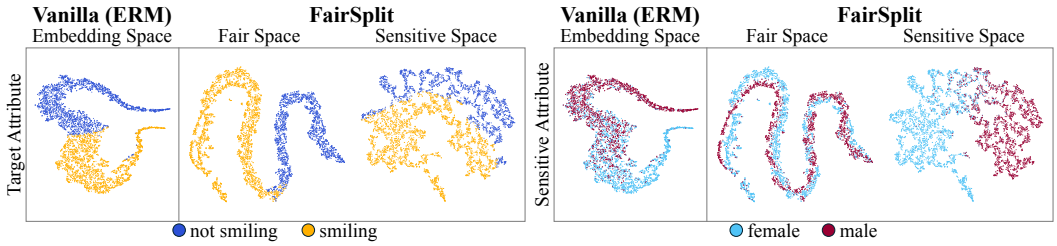


Figure 1: **Embedding spaces on CelebA-S**, where the task is to predict whether a person is smiling and gender is the sensitive attribute, i.e., the protected characteristic on which predictions should not depend, visualized with t-SNE and colored by the target (left) and the sensitive attribute (right). Each side shows three panels: the Vanilla embedding, trained by empirical risk minimization (ERM) without fairness constraints, and the Fair Space and Sensitive Space of FAIRSPLIT. In the Vanilla (ERM) embedding, *smiling*/*not smiling* correlates with *female*/*male*. FAIRSPLIT instead classifies from the Fair Space, which is guided towards independence from the sensitive attribute.

demographic parity gap is upper-bounded by both this dimensionality and the HSIC between the Fair Space and the sensitive attribute, both of which our method directly controls. The bound holds for multiclass classification and non-binary sensitive attributes, and applies to the downstream classifier operating on the learned representation. We validate FAIRSPLIT on standard benchmarks across multiple modalities, demonstrating competitive accuracy and strong fairness performance. FAIRSPLIT succeeds in multiclass settings with non-binary sensitive attributes, whereas many state-of-the-art baselines are limited to binary settings by design, a capability gap we highlight in Table 1.

Our main contributions are as follows:

- We propose FAIRSPLIT, an in-processing method for fair classification that decomposes the embedding into a Fair Space used for prediction and a Sensitive Space, learning the dimensionality of the Fair Space jointly with the classifier rather than fixing it in advance.
- We prove an upper bound on a continuous relaxation of the demographic parity gap that supports multiclass and non-binary sensitive attribute scenarios, and show that FAIRSPLIT controls this bound by reducing both the dimensionality of the Fair Space and the HSIC between the Fair Space and sensitive attributes.
- We empirically validate our theoretical results on established fair classification benchmarks, demonstrating strong performance in both binary settings and multiclass scenarios where most existing in-processing methods cannot be applied.

2 Related Work

Fairness Notions and Evaluation: Fairness is inherently multifaceted and depends on the perspective and application context. In this work, we focus on *group-level fairness*, with an emphasis on *demographic parity*, which requires the outcome of an ML model to be independent of a given sensitive attribute, so that group membership does not influence the prediction. A central and widely adopted measure for this is the demographic parity gap, ΔDP [14], which can be computed on model logits, prediction probabilities, or the final class assignments. Several recent works have proposed refinements and complementary measures, such as MCDP [28] and ABCC [21], which capture fairness violations continuously without reliance on an explicit decision threshold. Other prominent fairness notions include Equal Opportunity (EOpp) and Equalized Odds (EOdd) [23], which compare error rates across sensitive groups rather than prediction rates. We report all five metrics in our experiments, with demographic parity and its refinements (ABCC, MCDP) as our primary objectives.

Fair Classification and Representation Learning: Fairness-aware learning methods aim to satisfy one or more fairness notions while maintaining predictive performance. They are commonly divided into pre-processing, in-processing, and post-processing approaches. Pre-processing methods modify the training data, for example by reweighing samples or repairing features, so that a classifier trained on it is less biased [30, 14]. Post-processing methods instead adjust the predictions of a trained model [16, 1, 12], which often requires the sensitive attribute at test time [23].

In this work, we focus on *in-processing* methods, which add fairness objectives directly to training. There is a broad spectrum of in-processing approaches. In our experiments (Section 4), we compare our method to representative techniques from several major categories: adversarial approaches against a sensitive-attribute predictor (e.g., AdvDebias [55]), augmentation-based regularizers (e.g., FairMixup [8]), and gap-regularization methods (e.g., DiffMCDP [28]). Furthermore, we compare to DRAlign [37], which constrains intermediate layers by aligning decision rationales across groups, and several methods that, similar to FAIRSPLIT, enforce statistical independence between representations or predictions and sensitive attributes (e.g., MMD-B-Fair [11], HSIC(Y, S) [38], and PRemover [31]).

Table 1: **Method capabilities.** Support for multiclass targets and non-binary sensitive attributes ($G > 2$). ✓ supported, ✗ binary only, ♦ indicates that it should be possible, but has not been tested in the respective papers.

Method	Multiclass	$G > 2$
DRAlign [37]	✗	✗
DiffMCDP [28]	✗	✗
PRemover [31]	✗	✗
MMD-B-Fair [11]	✗	✗
FairMixup [8]	♦	✗
AdvDebias [55]	♦	♦
HSIC(Y, S) [38]	♦	♦
FAIRSPLIT (ours)	✓	✓

Independence and Space Splitting for Fairness: Our work relates to methods that learn representations explicitly designed to be independent of sensitive attributes. Early approaches such as LFR [54] and PRemover [31] discourage dependence via parity constraints and mutual information. Other methods enforce independence using statistical tests (e.g., MMD-B-Fair [11]), or the Rényi correlation [3]. The Hilbert-Schmidt Independence Criterion (HSIC) can detect non-linear, statistical dependencies [20]. Despite these properties, HSIC has only rarely been used for fair classification [22]. Existing HSIC-based approaches primarily enforce independence at the prediction level: HSIC(Y, S) [38] penalizes dependence between predicted labels and sensitive attributes, and FKL [45] applies HSIC in a regression setting with a focus on dimensionality reduction. Beyond classification, FairIB [53] uses HSIC in fair recommender systems. HSIC has also been used to decompose the embedding: building on the mask-based decomposition of Miklautz et al. [40], H-SPLID [41] uses HSIC to split the embedding into task-relevant and task-irrelevant features, and bounds the change in predictions under input perturbations by the dimensionality of the task-relevant space and its HSIC with the input. FAIRSPLIT brings this idea to fair classification, with a Fair Space used for prediction and a Sensitive Space that isolates sensitive information. A separate line of variational disentanglement methods [9, 50, 27] also decomposes the embedding space, but within a variational autoencoder aimed at fair generation or counterfactual synthesis rather than classification.

Open Challenges: Despite significant progress, several challenges in in-processing fair classification remain. As Table 1 shows, most existing methods are limited to binary sensitive attributes and binary classification, and do not naturally extend to multiclass or non-binary sensitive groups. Recent benchmarks [22] reflect this limitation. Yet multiclass classification and multi-group sensitive attributes are increasingly relevant in practice [1, 48, 49, 36, 12]. Addressing these settings in a principled manner remains an open problem, which we take up in this work.

3 Methodology

FAIRSPLIT trains a classifier together with a decomposition of its embedding into a Fair Space, which the classifier uses for prediction, and a Sensitive Space, which absorbs sensitive information. We first formalize the setting (Section 3.1) and then introduce the training objective that shapes the two spaces (Section 3.2). We next describe how the mask separating them is learned jointly with the network (Section 3.3), and finally bound the demographic parity gap of the resulting classifier (Section 3.4).

3.1 Problem Setup

We consider a classification task over a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i, s_i)\}_{i=1}^N$, where $\mathbf{x}_i \in \mathcal{X} \subseteq \mathbb{R}^d$ denotes the input, $y_i \in \mathcal{Y} = \{1, \dots, K\}$ the class label, and $s_i \in \mathcal{S} = \{1, \dots, G\}$ the sensitive attribute with G groups. We denote their one-hot encodings by $\mathbf{y}_i \in \{0, 1\}^K$ and $\mathbf{s}_i \in \{0, 1\}^G$. The model consists of an encoder $f_\phi : \mathcal{X} \to \mathcal{Z}$ that maps each input to an embedding $\mathbf{z}_i = f_\phi(\mathbf{x}_i)$ in the embedding space $\mathcal{Z} \subseteq \mathbb{R}^m$, and a classifier $g_\psi : \mathcal{Z} \to \mathbb{R}^K$ that outputs logits, from which class probabilities are obtained with a softmax. We denote all model parameters by $\theta = \{\phi, \psi\}$. For reference, Table 6 in Appendix A summarizes the notation used throughout the paper.

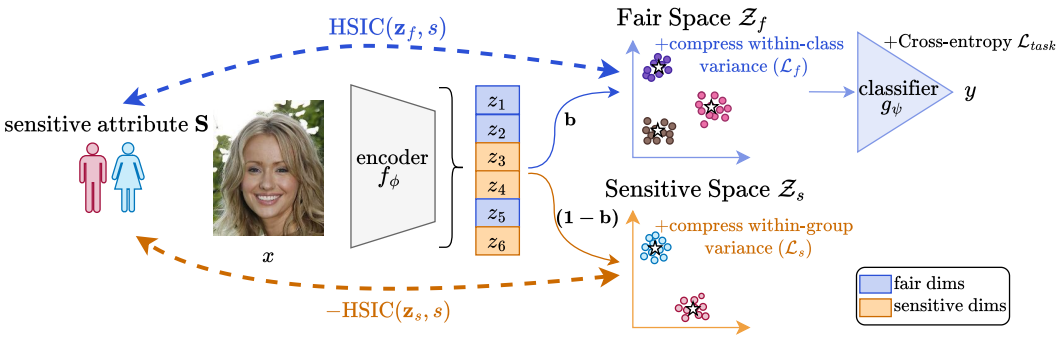


Figure 2: **Overview of FAIRSPLIT.** An encoder maps each input to an embedding $\mathbf{z}$, and a learned binary mask $\mathbf{b}$ splits it into a Fair Space $\mathbf{z}_f = \mathbf{b} \odot \mathbf{z}$ and a Sensitive Space $\mathbf{z}_s = (\mathbf{1} - \mathbf{b}) \odot \mathbf{z}$. Only the Fair Space is passed to the classifier. During training, HSIC regularization reduces the dependence between the Fair Space and the sensitive attribute and increases it in the Sensitive Space, which gives sensitive information a destination outside the classifier's input. The mask is updated in closed form by comparing, per dimension, how strongly the embedding varies within classes and within sensitive groups, so that the size of the Fair Space is learned end-to-end rather than fixed in advance.

3.2 Learning a Fair Decomposition

To achieve a decomposition of the embedding space into a Fair and a Sensitive Space while preserving high utility, we leverage three mechanisms: structural compression (Equation 2), dependence regularization via HSIC (Equation 4), and a utility objective for predictive performance (Equation 5). Figure 2 provides an overview of our approach. To decompose the embedding $\mathbf{z} = f_\phi(\mathbf{x})$, we introduce a binary mask $\mathbf{b} \in \{0, 1\}^m$. A binary mask provides a clear separation: each dimension belongs either to the Fair Space or the Sensitive Space. This enables the classifier to operate on a subset of features that is regularized toward independence from the sensitive attribute. The Fair Space is defined by $\mathbf{z}_f = \mathbf{b} \odot \mathbf{z}$, and the complementary Sensitive Space by $\mathbf{z}_s = (\mathbf{1} - \mathbf{b}) \odot \mathbf{z}$. The mask, and with it the dimensionality of the Fair Space $d_f = \|\mathbf{b}\|_0$, is learned during training rather than fixed in advance.

Structural Compression. Learning a binary mask requires a signal that indicates which dimensions should be assigned to which space. The structural compression loss (Equation 2) provides this signal by measuring how well each dimension aligns with class labels versus sensitive groups. Dimensions where within-class variance is low relative to within-group variance are assigned to the Fair Space. Conversely, dimensions with low within-group variance are assigned to the Sensitive Space.

We align the Fair Space with the class labels and the Sensitive Space with the sensitive attribute. This is achieved by regularizing the embedding space to compress class-relevant information into the Fair Space and sensitive information into the Sensitive Space (see Figure 2).

Let $\mathcal{I}_k$ and $\mathcal{I}_g$ denote the index sets for class k and sensitive group g, with centroids:

$$\boldsymbol{\mu}_k = \frac{1}{|\mathcal{I}_k|} \sum_{i \in \mathcal{I}_k} \mathbf{z}_i, \qquad \boldsymbol{\nu}_g = \frac{1}{|\mathcal{I}_g|} \sum_{i \in \mathcal{I}_g} \mathbf{z}_i. \tag{1}$$

The structural compression loss sums a Fair Space term $\mathcal{L}_f$ and a Sensitive Space term $\mathcal{L}_s$:

$$\mathcal{L}_{\text{comp}}(\mathcal{D}; \phi, \mathbf{b}) = \mathcal{L}_f + \mathcal{L}_s, \tag{2}$$

where $\mathcal{L}_f = \lambda_f \sum_k \sum_{i \in \mathcal{I}_k} \|\mathbf{b} \odot (\mathbf{z}_i - \boldsymbol{\mu}_k)\|_2^2$ and $\mathcal{L}_s = \lambda_s \sum_g \sum_{i \in \mathcal{I}_g} \|(\mathbf{1} - \mathbf{b}) \odot (\mathbf{z}_i - \boldsymbol{\nu}_g)\|_2^2$. The term $\mathcal{L}_f$ compresses within-class variance in the Fair Space, so the classifier learns to rely on class-specific information. The term $\mathcal{L}_s$ compresses within-group variance in the Sensitive Space to identify during training which features carry sensitive information and should be excluded from prediction. The hyperparameters λ_f and λ_s control the relative importance of each objective and influence the mask allocation.

HSIC Regularization. While the structural compression objective $\mathcal{L}_{\text{comp}}$ aligns the mask $\mathbf{b}$, it does not guarantee statistical independence. We address this by regularizing the learned subspaces

via the Hilbert-Schmidt Independence Criterion (HSIC) [20], a kernel-based measure of statistical dependence. For characteristic kernels (e.g., Gaussian RBF), $\mathrm{HSIC}(U, V) = 0$ iff $U \perp V$, making it well-suited for enforcing independence. Given n samples, HSIC can be estimated empirically as:

$$\widehat{\mathrm{HSIC}}(U, V) = (n-1)^{-2} \operatorname{tr}(\mathbf{K}_u \mathbf{H} \mathbf{K}_v \mathbf{H}), \tag{3}$$

where $\mathbf{K}_u, \mathbf{K}_v$ are kernel matrices and $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ is the centering matrix (see Appendix B for details). For a batch of n samples, let $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_n]^\top$ and $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_n]^\top$, and let $\mathbf{Z}_f$ and $\mathbf{Z}_s$ denote the masked versions of $\mathbf{Z}$, with rows $\mathbf{b} \odot \mathbf{z}_i$ and $(\mathbf{1} - \mathbf{b}) \odot \mathbf{z}_i$. We then apply HSIC with opposing objectives to the two spaces:

$$\mathcal{L}_{\text{reg}}(\mathcal{D}; \phi, \mathbf{b}) = \rho_f \widehat{\mathrm{HSIC}}(\mathbf{Z}_f, \mathbf{S}) - \rho_s \widehat{\mathrm{HSIC}}(\mathbf{Z}_s, \mathbf{S}) \tag{4}$$

The first term reduces statistical dependence between the Fair Space and the sensitive attribute s. The second term keeps sensitive information in $\mathbf{z}_s$, preventing it from leaking into the Fair Space.

Utility Objective. The decomposition serves a dual purpose during training: the model must simultaneously learn the classification task and separate fair from sensitive information. The Sensitive Space $\mathbf{z}_s$ is essential for this process. By learning an explicit space to capture sensitive information, it gives sensitive information a destination outside the Fair Space, reducing the dependence of $\mathbf{z}_f$ on the sensitive attribute. The classifier g_ψ operates solely on $\mathbf{z}_f$, so the full model is $h_\theta(\mathbf{x}) = g_\psi(\mathbf{b} \odot f_\phi(\mathbf{x}))$, and the utility objective is:

$$\mathcal{L}_{\text{task}}(\mathcal{D}; \theta, \mathbf{b}) = \frac{1}{N} \sum_{i=1}^{N} \ell(\mathbf{y}_i, g_\psi(\mathbf{b} \odot \mathbf{z}_i)), \tag{5}$$

where ℓ denotes the standard cross-entropy loss [19].

Total Objective. Combining the three components, we obtain the full training objective:

$$\mathcal{L}_{\text{total}}(\mathcal{D}; \theta, \mathbf{b}) = \mathcal{L}_{\text{task}} + \mathcal{L}_{\text{comp}} + \mathcal{L}_{\text{reg}} \tag{6}$$

Each component serves a distinct purpose: $\mathcal{L}_{\text{task}}$ maintains predictive performance, $\mathcal{L}_{\text{comp}}$ provides structural compression for the mask, $\mathcal{L}_{\text{reg}}$ guides the statistical dependence.

3.3 The FAIRSPLIT Algorithm

Since $\mathbf{b}$ is binary, the mask cannot be learned by gradient descent. We therefore optimize the model parameters and the mask alternately, as described in Algorithm 1 (Appendix E). First, we fix the mask and update the network parameters θ with Adam, while updating the centroids $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ with a moving average, so that the targets of the compression loss change smoothly across minibatches. Then, we fix the representations and update the mask by minimizing the two terms of $\mathcal{L}_{\text{comp}}$ with equal weights, so that the assignment of a dimension does not depend on how strongly λ_f and λ_s regularize the encoder toward the centroids. By relaxing the binary constraint to the interval $[0, 1]$, this sub-problem admits the following closed-form solution:

Proposition 3.1 (Closed-Form Mask Solution)**.**

$$b_j^* = \frac{V_j^{(s)}}{V_j^{(y)} + V_j^{(s)}}, \quad \forall j \in \{1, \ldots, m\}, \tag{7}$$

where $V_j^{(y)} = \sum_{k=1}^{K} \sum_{i \in \mathcal{I}_k} (z_{ij} - \boldsymbol{\mu}_{kj})^2$ *and* $V_j^{(s)} = \sum_{g=1}^{G} \sum_{i \in \mathcal{I}_g} (z_{ij} - \boldsymbol{\nu}_{gj})^2$ *are the within-class and within-group variances, and* z_{ij}, $\boldsymbol{\mu}_{kj}$, $\boldsymbol{\nu}_{gj}$ *are the* j*-th entries of* $\mathbf{z}_i$, $\boldsymbol{\mu}_k$, $\boldsymbol{\nu}_g$.

The derivation is in Appendix C. Since Equation 7 is evaluated on each minibatch and yields values in $[0, 1]$, we also smooth the mask with an exponential moving average and binarize it at 0.5.

3.4 Theoretical Analysis

We prove that a continuous relaxation of the demographic parity gap is bounded by the dimensionality of the Fair Space and the HSIC between the Fair Space and the sensitive attribute, for any number $G \geq 2$ of sensitive groups. For a model h_θ, we generalize the continuous relaxation of the demographic parity gap [54, 8, 21] to K classes and G groups:

$$\Delta_{\text{DP}}(h_\theta) := \max_{g,g'} \left\| \mathbb{E}[h_\theta(\mathbf{x}) \mid s = g] - \mathbb{E}[h_\theta(\mathbf{x}) \mid s = g'] \right\|_2. \tag{8}$$

It measures the largest Euclidean distance between the mean logits of any two sensitive groups. By operating on the continuous outputs, it captures bias in the model's behavior before a final class is assigned. In our experiments, ΔDP denotes the gap in predicted classes (Appendix F.2).

Theorem 3.2 (Demographic Parity Bound)**.** *Let $p_{\min} = \min_g \mathbb{P}(s = g) > 0$ denote the minimum group prevalence. Let $h_\theta = g_\psi \circ (\mathbf{b} \odot f_\phi)$ be a masked classifier where $f_\phi : \mathcal{X} \to \mathbb{R}^m$ is an encoder, $\mathbf{b} \in \{0,1\}^m$ is a binary mask with $d_f = \|\mathbf{b}\|_0$ active dimensions, and $g_\psi(\mathbf{z}) = \mathbf{W}\mathbf{z}$ is a linear classification head with K outputs and largest absolute entry $\|\mathbf{W}\|_{\max} \leq B$. Assume the Fair Space $\mathbf{z}_f = \mathbf{b} \odot f_\phi(\mathbf{x})$ satisfies $\mathbb{E}[\|\mathbf{z}_f\|_2] < \infty$. Then the demographic parity gap $\Delta_{DP}(h_\theta)$ is bounded as:*

$$\Delta_{DP}(h_\theta) \leq c_G B \sqrt{K d_f} \cdot \sqrt{\text{HSIC}(\mathbf{z}_f, \mathbf{s})}, \tag{9}$$

where HSIC *uses linear kernels on both arguments (Definition B.2) and $c_G \leq \sqrt{2}/p_{\min}$ is a constant that depends on the number of groups and their prevalences (Equation 16).*

Intuitively, Theorem 3.2 implies that demographic parity is jointly constrained by the statistical dependence ($\sqrt{\text{HSIC}}$) and the dimensionality of the Fair Space ($\sqrt{d_f}$). This motivates FAIRSPLIT's dual strategy: minimizing $\mathcal{L}_{\text{reg}}$ reduces the dependence between the Fair Space and the sensitive attribute, while the learnable mask $\mathbf{b}$ moves dimensions that carry sensitive information to the Sensitive Space, reducing d_f. In practice, $\mathcal{L}_{\text{reg}}$ estimates HSIC with a Gaussian kernel: since the kernel is characteristic, driving the first term of $\mathcal{L}_{\text{reg}}$ to zero enforces $\mathbf{z}_f \perp s$, under which the dependence term in Equation 9 vanishes. The fairness-utility trade-off is empirically stable across smooth kernel families and across a range of bandwidths (see Appendix G.10). We observe that a smaller d_f goes along with better fairness across multiple metrics in Figure 4.

Proof Sketch. The proof (Appendix D) proceeds by first linking the demographic parity gap to the HSIC of the model outputs (Lemma D.1). We then exploit the masked linear classification head structure $h_\theta(\mathbf{x}) = \mathbf{W}(\mathbf{b} \odot \mathbf{z})$ to decouple this dependence into a weight norm term and a representation covariance term. The factor $\sqrt{d_f}$ arises from bounding the Frobenius norm of $\mathbf{W}$ restricted to active dimensions. Since every quantity in the bound can be computed from the trained model and the data, it can be evaluated after training (Appendix G.1).

3.5 Limitations and Discussion

Our theoretical bound applies to classifiers with linear heads, covering standard architectures like ResNets [25], ViTs [13], and MLPs, while extending guarantees to non-linear heads remains open. In line with this linear scope, we observe that a non-linear probe can still recover the sensitive attribute from the Fair Space (Appendix G.4). Empirically, however, FAIRSPLIT retains substantial fairness gains when the linear head is replaced by a multi-layer non-linear classifier on CelebA-A and UTKFace-AR (Appendix G.5). The linear-head assumption applies only to the classifier: the encoder f_ϕ can be arbitrarily non-linear. FAIRSPLIT applies to fine-tuned ViTs and frozen foundation models (Section 4.3) and transfers across five pretrained encoders spanning dimensionalities from 512 to 2048 without per-architecture retuning, with the mask adapting d_f to each embedding size (Appendix G.2). Our bound scales with the inverse of the minimum group prevalence $p_{\min}$, reflecting the known challenge of fairness under imbalance [5], yet FAIRSPLIT retains its advantage on intersectional attributes with $p_{\min} = 0.05$ (Table 3) and under imposed imbalance (Appendix G.6). More generally, when the target and the sensitive attribute are dependent, any classifier that reduces the demographic parity gap below the difference between the groups' base rates must incur a minimum error [58], a trade-off we quantify for each benchmark in Section 4.1. A practical limitation is the requirement for sensitive attributes during training, though FAIRSPLIT is partly robust to noisy or missing sensitive labels (Appendix G.12), and future work could explore semi-supervised approaches when annotations are unavailable. Finally, the Sensitive Space retains information about the sensitive attribute, so applications that must keep it private should protect the model parameters.

4 Experiments

We follow the evaluation protocol of the in-processing fair classification benchmark by Han et al. [22], averaging experiments over five seeds each, and extend it to accommodate scenarios with multiclass classification and non-binary sensitive attributes. Code is available at `https://github.com/shkabrii/fairsplit`.

Datasets. For binary classification experiments, we use the Adult [35] tabular dataset and the CelebA-A [39] image dataset. In Adult, the task is to predict whether an individual's income exceeds \$50K, with gender (male/female) as the binary sensitive attribute. In CelebA-A, we predict the attractive attribute from face images, using gender as the binary sensitive attribute. To evaluate performance in multiclass settings with non-binary sensitive attributes, we use UTKFace-AR [56], where the task is to classify age into three groups (Young: $\leq$25, Middle-aged: 26-45, and Old: $>$45) from face images, with race as the sensitive attribute grouped into three categories (White, Black, and Other). Finally, to evaluate intersectional fairness under severe imbalance, we use the COMPAS recidivism dataset [2] with an intersectional sensitive attribute (gender $\times$ age) yielding four groups with $p_{\min} = 0.05$. For the experiments with large pretrained models (Section 4.3), we additionally use FairFace [32] and CelebA [39] at 224×224 resolution. Further details are provided in Appendix F.1.

Comparison Methods. We compare FAIRSPLIT against several representative in-processing fair classification baselines: empirical risk minimization (ERM), the adversarial approach AdvDebias [55], the augmentation-based method FairMixup [8], the alignment method DRAlign [37], the independence-based methods HSIC(Y, S) [38], PRemover [31], and MMD-B-Fair [11], as well as the gap-regularization method DiffMCDP [28]. We note that many of these methods are restricted to binary classification with binary sensitive attributes by design (see Table 1).

Evaluation Metrics. We report Average Precision (AP↑) for utility in binary settings and F1 score (F1↑) for multiclass settings, along with several fairness metrics, including demographic parity gap (ΔDP↓), equal opportunity (ΔEOpp↓), equalized odds (ΔEOdd↓), ABCC↓ [21], and MCDP↓ [28]. Our theoretical guarantee (Theorem 3.2) concerns demographic parity, while ΔEOpp and ΔEOdd compare error rates rather than prediction rates across groups and are not covered by it, so we report them as complementary empirical metrics. For multiclass targets and non-binary sensitive attributes, we extend these metrics by taking the maximum disparity across all classes and all pairs of sensitive groups, following Denis et al. [12] and Alghamdi et al. [1]. Note that in fair classification, predictive accuracy is not the sole criterion. When training labels reflect historical biases, a model that perfectly reproduces them may be accurate yet unfair [38]. Hence, we aim to balance accuracy and fairness by considering only models that achieve at least 95% of ERM's validation utility (AP/F1) and comparing their fairness, following prior work [29, 28]. See Appendix F.2 for more details. Following Han et al. [22], to characterize the utility-fairness trade-off rather than a single operating point, we additionally visualize trade-off curves by varying each method's fairness-controlling hyperparameter (see Figure 3). Each curve connects a method's points in order of this hyperparameter, so a curve changes direction wherever a fairness metric does not vary monotonically with it.

4.1 Comparison and Trade-Offs in Binary Settings

Although most baselines are designed for binary classification with binary sensitive attributes, FAIRSPLIT matches or exceeds them on both datasets (Table 2, Figure 3). FAIRSPLIT achieves the lowest ABCC on Adult and CelebA-A and the lowest MCDP on CelebA-A. A low ABCC means the per-group score distributions are close on average across decision thresholds, while MCDP measures the gap at the single worst threshold. On Adult, predicted scores cluster in a narrow range below 0.5, so small per-seed differences inflate MCDP while ABCC stays small, which is why methods regularizing predictions directly lead on MCDP there. On ΔEOpp and ΔEOdd, FAIRSPLIT achieves the best results on both datasets. While theoretical trade-offs exist between demographic parity and equalized odds [34], FAIRSPLIT attains a more efficient trade-off in our experiments. More broadly, part of the fairness-utility trade-off is inherent to the data, because it depends on how much the groups' base rates differ. We denote this gap by $\Delta\pi = |\mathbb{P}(y{=}1 \mid s{=}1) - \mathbb{P}(y{=}1 \mid s{=}2)|$ and report it for each benchmark in Tables 2 and 3. Zhao and Gordon [57, 58] show that a classifier with a ΔDP below $\Delta\pi$ cannot have a low error on both groups, since its two group-wise error rates sum to at least $\Delta\pi - \Delta$DP. Once ΔDP exceeds $\Delta\pi$, this lower bound becomes negative and no longer constrains the

Table 2: **Performance on binary settings.** CelebA-A and Adult are binary classification tasks with gender as the sensitive attribute, and we measure utility by average precision (AP). For each method, we consider all configurations that reach at least 95% of ERM's validation AP and report the one with the lowest ΔDP, without further optimizing AP. **Bold** and underline mark the best and second-best results. Results are averaged over 5 seeds. $\Delta\pi$ is the base-rate gap between groups (Section 4.1).

	CelebA-A ($\Delta\pi = 40.0$)						Adult ($\Delta\pi = 19.8$)					
	ΔDP $\downarrow$	ABCC $\downarrow$	MCDP $\downarrow$	ΔEOpp $\downarrow$	ΔEOdd $\downarrow$	AP $\uparrow$	ΔDP $\downarrow$	ABCC $\downarrow$	MCDP $\downarrow$	ΔEOpp $\downarrow$	ΔEOdd $\downarrow$	AP $\uparrow$
ERM	52.8±0.8	35.9±0.8	53.4±0.8	36.8±1.5	75.0±2.1	84.5±0.4	19.4±0.6	18.7±0.2	36.6±0.6	9.3±2.0	18.2±2.3	**76.7±0.5**
AdvDebias	45.9±3.4	31.2±1.6	46.8±3.8	26.0±4.4	57.1±7.6	**84.9±0.7**	6.1±5.7	7.7±5.0	16.7±11.1	21.8±9.5	25.3±8.9	72.8±2.6
FairMixup	37.2±1.1	19.5±0.9	39.5±1.2	19.5±2.4	39.8±2.5	83.1±0.1	1.1±0.6	4.6±0.5	17.2±2.4	32.6±2.8	36.0±3.3	72.9±0.4
DRAlign	31.7±4.9	15.4±2.9	33.6±4.3	12.4±4.8	29.9±9.1	80.3±1.2	**0.7±0.5**	6.1±0.3	19.8±0.9	34.2±1.0	37.8±1.0	73.6±0.4
HSIC(Y, S)	23.9±1.4	13.6±0.5	27.0±1.7	3.3±1.5	11.6±3.0	81.3±0.5	1.9±0.5	4.1±0.3	11.2±1.1	27.8±1.7	31.1±2.0	73.0±0.4
DiffMCDP	20.4±2.4	14.4±1.6	22.3±2.2	3.2±1.0	9.8±2.5	80.6±1.0	3.0±0.4	4.3±0.4	**11.0±1.1**	25.8±1.6	28.4±1.7	73.6±0.2
PRemover	25.1±1.2	13.4±0.5	26.7±1.2	3.1±1.7	13.5±3.0	80.6±0.2	3.6±0.5	7.1±0.4	19.1±1.1	24.7±1.3	26.9±1.6	74.5±0.4
MMD-B-F.	37.1±13.3	24.2±10.5	37.8±12.9	17.9±13.6	42.7±25.0	80.4±4.6	3.7±0.8	5.2±1.7	15.0±5.5	24.7±2.8	26.8±3.4	73.1±1.0
FAIRSPLIT	**18.3±1.1**	**9.4±1.3**	**19.6±1.8**	**2.6±0.3**	**5.2±1.7**	80.4±1.1	1.0±0.9	**2.4±0.7**	19.8±4.5	**8.2±4.1**	**8.3±4.2**	72.9±0.9

Table 3: **Performance on multiclass and intersectional settings.** UTKFace-AR has a 3-class age target and 3 racial groups, with F1 as utility. COMPAS has a binary target and an intersectional gender $\times$ age attribute, with AP as utility. Methods restricted to binary targets or sensitive attributes are omitted (Table 1). As in Table 2, all configurations achieve at least 95% of ERM's validation utility, and **bold** and underline mark the best and second-best results. Results are averaged over 5 seeds. With more than two groups, $\Delta\pi$ is the largest base-rate gap over pairs of groups and classes.

	UTKFace-AR ($\Delta\pi = 25.9$)						COMPAS ($\Delta\pi = 26.6$)					
	ΔDP $\downarrow$	ABCC $\downarrow$	MCDP $\downarrow$	ΔEOpp $\downarrow$	ΔEOdd $\downarrow$	F1 $\uparrow$	ΔDP $\downarrow$	ABCC $\downarrow$	MCDP $\downarrow$	ΔEOpp $\downarrow$	ΔEOdd $\downarrow$	AP $\uparrow$
ERM	27.9±1.4	27.1±2.6	36.2±2.4	29.1±5.2	35.9±3.0	73.8±0.7	35.2±3.3	17.9±1.8	58.8±4.5	24.9±4.7	55.4±8.7	**69.1±2.4**
AdvDebias	28.7±3.0	21.1±1.9	32.9±2.5	32.7±7.8	38.5±7.9	**74.7±0.8**	26.4±7.7	13.9±3.1	47.8±10.9	29.5±5.7	44.7±12.4	68.4±2.1
HSIC(Y, S)	**11.9±1.3**	12.0±0.8	20.4±2.4	18.1±2.7	24.1±3.5	70.3±1.1	20.5±3.1	6.7±0.7	35.8±4.6	28.3±9.2	36.1±9.8	67.9±2.3
FAIRSPLIT	12.6±1.1	**9.1±1.0**	**18.7±1.2**	**14.9±4.6**	**22.1±4.9**	70.4±1.5	**16.8±3.6**	**1.2±0.4**	**32.9±6.2**	**23.8±6.7**	**31.5±7.6**	66.8±2.4

classifier. On CelebA-A ($\Delta\pi = 40.0$), ERM and AdvDebias both have a ΔDP above the gap (52.8 and 45.9), so the bound does not prevent AdvDebias from improving fairness and AP at the same time. FAIRSPLIT instead reduces ΔDP to 18.3, well below the gap, which the bound allows only at a minimum error. In a controlled experiment that changes only $\Delta\pi$, the AP cost of FAIRSPLIT increases from 0.9 to 6.7 as $\Delta\pi$ grows from 5 to 45 points (Appendix G.7).

4.2 Multiclass, Non-Binary, and Intersectional Settings

The embedding-decomposition mechanism behind FAIRSPLIT is agnostic to the number of classes K and groups G, so the same algorithm applies to multiclass targets and non-binary sensitive attributes without modification, a property only AdvDebias and HSIC(Y, S) share among our baselines. Table 3 reports results on UTKFace-AR and COMPAS, the latter being the regime where Theorem 3.2's $1/p_{\min}$ factor makes the bound loosest. On both, FAIRSPLIT outperforms on most fairness metrics and is competitive on the rest, with utility within the 95%-of-ERM threshold. On COMPAS, FAIRSPLIT also attains the lowest ABCC (1.2 against 6.7 for the next-best method). The $1/p_{\min}$ factor stems from the pairwise definition of demographic parity (Lemma D.1), yet FAIRSPLIT's empirical fairness does not degrade at the rate the bound predicts. The imposed-imbalance experiment in Appendix G.6 confirms this gap between guarantee and behavior persists across imbalance regimes.

4.3 Vision Transformers and Frozen Foundation Models

FAIRSPLIT applies unchanged to large pretrained models (Table 4). When FAIRSPLIT fine-tunes a ViT-B/16 [13] end-to-end on FairFace at 224×224, it lowers ΔDP from 23.0 to 9.5 and more than halves every other fairness metric, while retaining 97% of ERM's AP. We also apply FAIRSPLIT to three frozen backbones, CLIP ViT-L/14, CLIP ResNet-50, and DINOv2 ViT-L/14 [46, 25, 43], training a small projection on top of the features and learning the Fair Space for one CelebA attribute. A new linear head trained on this Fair Space, without sensitive labels, then predicts a related, unseen attribute, and improves every fairness metric on all three backbones while retaining at least 95% of ERM's AP. Appendix G.3 describes the setup and reports results for both tasks.

Table 4: **FAIRSPLIT on large pretrained models** (5 seeds, gender as the sensitive attribute). ERM uses the same architecture as FAIRSPLIT. ViT-B/16 is fine-tuned end-to-end on FairFace to predict whether a person is at least 30 years old. For the frozen backbones, the Fair Space is learned for predicting high cheekbones on CelebA. A new linear head is then trained on this fixed Fair Space to predict a different attribute, age. d_f/m is the size of the Fair Space relative to the full embedding.

Backbone	Method	AP ↑	ΔDP ↓	ABCC ↓	ΔEOpp ↓	ΔEOdd ↓	d_f/m
ViT-B/16	ERM	**88.9±0.2**	23.0±0.5	17.2±0.1	22.3±1.1	33.5±1.1	768 / 768
(fine-tuned)	FAIRSPLIT	86.4±0.3	**9.5±1.6**	**5.8±1.4**	**8.1±2.6**	**9.5±1.9**	552 / 768
CLIP ViT-L/14	ERM	**96.5±0.1**	21.4±0.9	20.0±0.6	4.9±0.5	23.6±2.4	256 / 256
(frozen)	FAIRSPLIT	93.1±0.4	**9.1±1.3**	**6.4±0.7**	**0.8±0.3**	**4.8±1.6**	174 / 256
CLIP ResNet-50	ERM	**96.6±0.1**	23.4±0.4	21.0±0.3	6.1±0.5	29.4±1.5	256 / 256
(frozen)	FAIRSPLIT	93.6±0.4	**12.2±2.4**	**7.9±1.5**	**0.9±0.5**	**8.5±5.0**	168 / 256
DINOv2 ViT-L/14	ERM	**95.2±0.1**	23.5±0.4	22.1±0.2	7.3±0.3	35.3±1.7	256 / 256
(frozen)	FAIRSPLIT	90.9±0.5	**9.1±1.8**	**5.9±1.2**	**0.5±0.4**	**7.1±1.7**	166 / 256

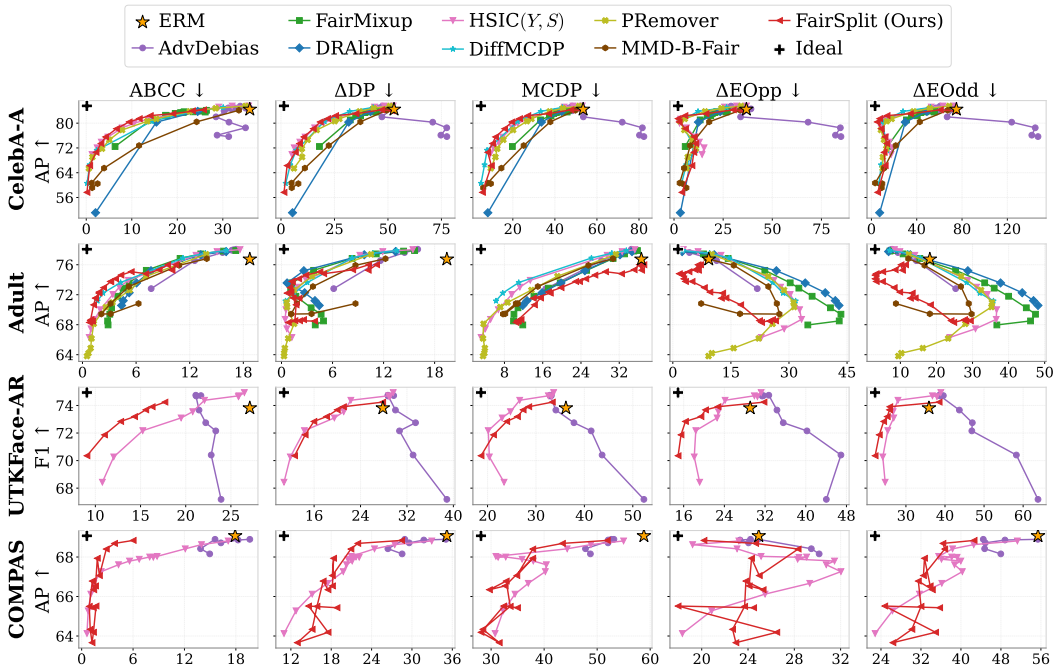


Figure 3: **Trade-off between performance and fairness** (higher performance is better) across four datasets. Each row is one dataset (CelebA-A, Adult, UTKFace-AR, COMPAS) with its own y-axis (AP for binary classification, F1 for the multiclass UTKFace-AR), and each column is one fairness metric. ERM (★) is a single point. The ideal model (✚) lies in the top-left corner of every panel. FAIRSPLIT matches or outperforms competing methods across most settings and achieves the best ABCC on every dataset in Tables 2 and 3.

4.4 Controlling Fairness via the Sensitive Space

Theorem 3.2 establishes a connection between the dimensionality of the Fair Space d_f and fairness: a smaller d_f tightens the upper bound on the demographic parity gap. To validate this empirically, we train on CelebA-A with varying ρ_s, which controls how strongly sensitive information is pushed into the Sensitive Space. Figure 4 aligns with our theoretical finding. Higher ρ_s yields smaller d_f, and models with lower d_f generally achieve better fairness, with the same trend visible on the threshold-free metrics ABCC and MCDP. Maximizing the dependence between $\mathbf{z}_s$ and the sensitive attribute via ρ_s encourages the model to move sensitive information out of the Fair Space, so the Fair Space becomes smaller without fixing d_f in advance, which tightens the bound in Theorem 3.2.

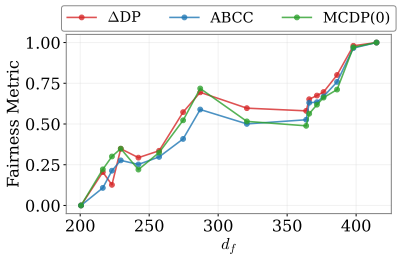


Figure 4: **Fairness vs. Fair Space dim.** Smaller d_f generally goes along with better fairness across ΔDP, ABCC, and MCDP. Metrics are normalized to $[0, 1]$.

Table 5: **Ablation of the Sensitive Space** (CelebA-A, 5 seeds). Penalizing HSIC on the full embedding (*No Sensitive Space*) halves ERM's ΔDP, adding the mask without the Sensitive Space losses (*Unregularized*, $\lambda_s = \rho_s = 0$) reduces it further, and full FAIRSPLIT yields the lowest ΔDP and ABCC with the smallest Fair Space at comparable AP. More metrics and datasets are in Table 20 in the appendix.

Variant	AP ↑	ΔDP ↓	ABCC ↓	d_f
ERM	**84.5±0.4**	52.8±0.8	35.9±0.8	512
No Sensitive Space	80.7±0.7	26.6±1.4	16.1±1.0	512
Unregularized	80.1±0.8	22.2±2.4	12.4±1.7	391
FAIRSPLIT	80.4±1.1	**18.3±1.1**	**9.4±1.3**	334

4.5 Ablations and Hyperparameter Sensitivity

Why an explicit Sensitive Space? A natural question is whether the Sensitive Space is necessary, or whether suppressing dependence on s in a single embedding via HSIC would suffice. Table 5 answers this by introducing the components of FAIRSPLIT one at a time on CelebA-A. When HSIC is applied to the full embedding without any decomposition, it acts on the entire input of the classifier, so sensitive information can only be suppressed, not moved to a separate space. This lowers ΔDP from 52.8 for ERM to 26.6, which remains well above the 18.3 of FAIRSPLIT. Learning the mask separates the dimensions that mainly encode the target from those that mainly encode the sensitive attribute, so the classifier only sees the Fair Space. This already improves fairness, before any loss term acts on the Sensitive Space ($\lambda_s = \rho_s = 0$). Regularizing the Sensitive Space then actively moves sensitive information into it, which shrinks the Fair Space further and gives the lowest ΔDP at comparable AP. This matches the role of d_f in Theorem 3.2 as a structural regularizer of the bound, and probes confirm that the Sensitive Space retains the sensitive attribute (Appendix G.4). The same pattern holds on UTKFace-AR (Appendix G.9).

How sensitive is FAIRSPLIT to its hyperparameters? We vary each of the four loss weights on CelebA-A while keeping the others fixed (Appendix G.8, Figure 6). Fairness depends mostly on the independence weight ρ_f: a larger ρ_f clearly improves ΔDP, ABCC, and MCDP, while AP decreases only gradually. The weight λ_f of the Fair Space has little effect on the results. The two weights of the Sensitive Space, λ_s and ρ_s, mainly decide how many dimensions the mask assigns to the Sensitive Space, and larger values leave fewer dimensions in the Fair Space. Finally, the results remain stable for different kernel types and for different kernel bandwidths (Appendix G.10). FAIRSPLIT also tolerates partly missing sensitive labels: on CelebA-A, ΔDP rises to 28.0 with 30% of labels missing, versus 52.8 for ERM, while incorrect labels are harder, and 30% of them remove the gain (Appendix G.12).

5 Conclusion and Future Work

We propose FAIRSPLIT, a fair classification method that decomposes the embedding space into two subspaces: the Sensitive Space provides an explicit destination for sensitive information during training, while the Fair Space is isolated for prediction. Unlike methods that regularize predictions directly, FAIRSPLIT regularizes dependence at the representation level via HSIC, enabling a theoretical guarantee on demographic parity. Our experiments demonstrate fairness improvements across multiple metrics, including in multiclass settings with non-binary sensitive attributes, and these improvements extend to a fine-tuned ViT-B/16 and to frozen foundation models, where a Fair Space learned for one task keeps its fairness gains on a related, unseen task (Section 4.3). By maintaining an isolated Sensitive Space rather than discarding sensitive information, FAIRSPLIT provides additional flexibility in choosing whether sensitive information should be included or discarded depending on the downstream task. A natural next step is to extend the theoretical guarantee beyond linear heads, using the characteristic kernels that FAIRSPLIT already applies in its HSIC regularization.

Acknowledgments and Disclosure of Funding

We want to thank the open source community, particularly the developers of PyTorch [44], NumPy [24], scikit-learn [6], and Matplotlib [26], and the authors of FFB [22], MCDP [28], and MMD-B-Fair [11] for releasing their code.

References

[1] W. Alghamdi, H. Hsu, H. Jeong, H. Wang, P. Michalak, S. Asoodeh, and F. Calmon. Beyond adult and compas: Fair multi-class prediction via information projection. *Advances in Neural Information Processing Systems*, 35:38747–38760, 2022.

[2] J. Angwin, J. Larson, S. Mattu, and L. Kirchner. Machine bias: There's software used across the country to predict future criminals. and it's biased against blacks. `https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing`, 2016. ProPublica.

[3] S. Baharlouei, M. Nouiehed, A. Beirami, and M. Razaviyayn. Rényi fair inference. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=HkgsUJrtDB`.

[4] B. Becker and R. Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

[5] D. Brzezinski, J. Stachowiak, J. Stefanowski, I. Szczech, R. Susmaga, S. Aksenyuk, U. Ivashka, and O. Yasinskyi. Properties of fairness measures in the context of varying class imbalance and protected group ratios. *ACM Transactions on Knowledge Discovery from Data*, 18(7):1–18, 2024.

[6] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[7] E. Caldeira, J. S. Cardoso, A. F. Sequeira, and P. C. Neto. Mst-kd: Multiple specialized teachers knowledge distillation for fair face recognition. In *European Conference on Computer Vision*, pages 211–228. Springer, 2024.

[8] C.-Y. Chuang and Y. Mroueh. Fair mixup: Fairness via interpolation. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=DNl5s5BXeBn`.

[9] E. Creager, D. Madras, J.-H. Jacobsen, M. Weis, K. Swersky, T. Pitassi, and R. Zemel. Flexibly fair representation learning by disentanglement. In *International conference on machine learning*, pages 1436–1445. PMLR, 2019.

[10] S. Dehdashtian, L. Wang, and V. Boddeti. Fairerclip: Debiasing clip's zero-shot predictions using functions in rkhss. In *International Conference on Learning Representations*, volume 2024, pages 26185–26202, 2024.

[11] N. Deka and D. J. Sutherland. Mmd-b-fair: Learning fair representations with statistical testing. In *International Conference on Artificial Intelligence and Statistics*, pages 9564–9576. PMLR, 2023.

[12] C. Denis, R. Elie, M. Hebiri, and F. Hu. Fairness guarantees in multi-class classification with demographic parity. *Journal of Machine Learning Research*, 25(130):1–46, 2024. URL `http://jmlr.org/papers/v25/23-0322.html`.

[13] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[14] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 259–268, 2015.

[15] Q. Feng, M. Du, N. Zou, and X. Hu. Fair machine learning in healthcare: A survey. *IEEE Transactions on Artificial Intelligence*, 2024.

[16] K. Fukuchi. Meta optimality for demographic parity constrained regression via post-processing. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=iuDS9qDzJG.

[17] K. Fukumizu, A. Gretton, X. Sun, and B. Schölkopf. Kernel measures of conditional dependence. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/3a0772443a0739141292a5429b952fe6-Paper.pdf.

[18] S. Gong, X. Liu, and A. K. Jain. Jointly de-biasing face recognition and demographic attribute estimation. In *European conference on computer vision*, pages 330–347. Springer, 2020.

[19] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT Press, 2016.

[20] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pages 63–77. Springer, 2005.

[21] X. Han, Z. Jiang, H. Jin, Z. Liu, N. Zou, Q. Wang, and X. Hu. Retiring Δ dp: New distribution-level metrics for demographic parity. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=LjDFIWWVVa.

[22] X. Han, J. Chi, Y. Chen, Q. Wang, H. Zhao, N. Zou, and X. Hu. Ffb: A fair fairness benchmark for in-processing group fairness methods. In *The Twelfth International Conference on Learning Representations*, 2024.

[23] M. Hardt, E. Price, and N. Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016.

[24] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, Sept. 2020. doi: 10.1038/s41586-020-2649-2. URL https://doi.org/10.1038/s41586-020-2649-2.

[25] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[26] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9 (3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

[27] T. Jang and X. Wang. Fades: Fair disentanglement with sensitive relevance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12067–12076, June 2024.

[28] J. Jin, H. Li, and F. Feng. On the maximal local disparity of fairness-aware classifiers. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=5cm2jGct2W.

[29] S. Jung, T. Park, S. Chun, and T. Moon. Re-weighting based group fairness regularization via classwise robust optimization. *arXiv preprint arXiv:2303.00442*, 2023.

[30] F. Kamiran and T. Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.

[31] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 35–50. Springer, 2012.

[32] K. Karkkainen and J. Joo. Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1548–1558, 2021.

[33] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. doi: 10.48550/arXiv.1412.6980.

[34] J. Kleinberg, S. Mullainathan, and M. Raghavan. Inherent Trade-Offs in the Fair Determination of Risk Scores. In C. H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 43:1–43:23, Dagstuhl, Germany, 2017. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-029-3. doi: 10.4230/LIPIcs.ITCS.2017.43. URL https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITCS.2017.43.

[35] R. Kohavi et al. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *Kdd*, volume 96, pages 202–207, 1996.

[36] F. Laakom, H. Chen, J. Schmidhuber, and Y. Bu. Fairness overfitting in machine learning: An information-theoretic perspective. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=saBXnGIDSj.

[37] T. Li, Q. Guo, A. Liu, M. Du, Z. Li, and Y. Liu. Fairer: fairness as decision rationale alignment. In *International Conference on Machine Learning*, pages 19471–19489. PMLR, 2023.

[38] Z. Li, A. Pérez-Suay, G. Camps-Valls, and D. Sejdinovic. Kernel dependence regularizers and gaussian processes with applications to algorithmic fairness. *Pattern Recognition*, 132:108922, 2022.

[39] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[40] L. Miklautz, L. Bauer, D. Mautz, S. Tschiatschek, C. Böhm, and C. Plant. Details (don't) matter: Isolating cluster information in deep embedded spaces. In *IJCAI International Joint Conference on Artificial Intelligence 2021*, pages 2826–2832. International Joint Conferences on Artificial Intelligence, 2021.

[41] L. Miklautz, C. Shi, A. Shkabrii, T. T. Davarakis, P. Lam, C. Plant, J. Dy, and S. Ioannidis. H-SPLID: HSIC-based saliency preserving latent information decomposition. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=0owHzJrnB6.

[42] P. C. Neto, N. Damer, J. S. Cardoso, and A. F. Sequeira. Balancing beyond discrete categories: Continuous demographic labels for fair face recognition. *arXiv preprint arXiv:2506.01532*, 2025.

[43] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[44] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. *CoRR*, abs/1912.01703, 2019. doi: 10.48550/ARXIV.1912.01703. URL https://doi.org/10.48550/arXiv.1912.01703.

[45] A. Pérez-Suay, V. Laparra, G. Mateo-García, J. Muñoz-Marí, L. Gómez-Chova, and G. Camps-Valls. Fair kernel learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 339–355. Springer, 2017.

[46] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

[47] M. Raghavan, S. Barocas, J. Kleinberg, and K. Levy. Mitigating bias in algorithmic hiring: Evaluating claims and practices. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 469–481, 2020.

[48] J. Rouzot, J. Ferry, and M.-J. Huguet. Learning optimal fair scoring systems for multi-class classification. In *2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 197–204. IEEE, 2022.

[49] S. Sabato, E. Treister, and E. Yom-Tov. Disparate conditional prediction in multiclass classifiers. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=Jf8MPTOpLc.

[50] M. H. Sarhan, N. Navab, A. Eslami, and S. Albarqouni. Fairness by learning orthogonal disentangled representations. In *European Conference on Computer Vision*, pages 746–761. Springer, 2020.

[51] M. Wang, W. Deng, J. Hu, X. Tao, and Y. Huang. Racial faces in the wild: Reducing racial bias by information maximization adaptation network. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 692–702. IEEE, 2019.

[52] Z. Wang, T. Jian, A. Masoomi, S. Ioannidis, and J. Dy. Revisiting hilbert-schmidt information bottleneck for adversarial robustness. *Advances in Neural Information Processing Systems*, 34: 586–597, 2021.

[53] J. Xie, Y. Yang, Z. Wang, and L. Wu. Learning fair representations for recommendation via information bottleneck principle. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 2469–2477, 2024.

[54] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning fair representations. In *International conference on machine learning*, pages 325–333. PMLR, 2013.

[55] B. H. Zhang, B. Lemoine, and M. Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018.

[56] Z. Zhang, Y. Song, and H. Qi. Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5810–5818, 2017.

[57] H. Zhao and G. Gordon. Inherent tradeoffs in learning fair representations. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/b4189d9de0fb2b9cce090bd1a15e3420-Paper.pdf.

[58] H. Zhao and G. J. Gordon. Inherent tradeoffs in learning fair representations. *Journal of Machine Learning Research*, 23(57):1–26, 2022. URL http://jmlr.org/papers/v23/21-1427.html.

Appendix

A Notation Summary **16**

B Technical Preliminaries **17**

B.1 Hilbert-Schmidt Independence Criterion (HSIC) 17

C Proof of Proposition 3.1 (Closed-Form Mask) **17**

D Proof of Theorem 3.2 (Main Result) **18**

D.1 Connecting the Demographic Parity to HSIC 18

D.2 Proof of the Main Theorem 19

E Algorithm Pseudocode **20**

F Reproducibility and Experimental Details **20**

F.1 Datasets 20

F.2 Metrics 21

F.3 Implementation 21

F.4 Software and Hardware Setup 22

F.5 Hyperparameters 22

G Additional Results **23**

G.1 Bound Tightness 23

G.2 Robustness Across Encoders and Embedding Dimensionalities 23

G.3 Fine-Tuned Vision Transformers and Frozen Foundation Models 24

G.4 Probing the Fair and Sensitive Spaces 25

G.5 Empirical Performance with Non-linear Classifier Heads 26

G.6 Performance under Imposed Group Imbalance 27

G.7 Fairness-Utility Trade-off and the Base-Rate Gap 27

G.8 Hyperparameter Sensitivity 27

G.9 Component Ablation Across Datasets 28

G.10 HSIC Kernel and Bandwidth Sensitivity 29

G.11 Computational Cost 29

G.12 Robustness to Noisy and Missing Sensitive Attributes 29

A Notation Summary

Table 6: Summary of notation used in this paper.

Symbol	Description
Data and Labels	
$\mathbf{x}, \mathbf{x}_i$	Input features
y, y_i	Class label $\in \{1, \ldots, K\}$
$\mathbf{y}, \mathbf{y}_i$	One-hot encoding of class label
s, s_i	Sensitive attribute $\in \{1, \ldots, G\}$
$\mathbf{s}, \mathbf{s}_i$	One-hot encoding of sensitive attribute
s_g	Indicator $\mathbb{1}[s = g]$ for group g
Spaces, Sets, and Dimensions	
$\mathcal{X}$	Input space $\subseteq \mathbb{R}^d$
$\mathcal{Z}$	Embedding space $\subseteq \mathbb{R}^m$
$\mathcal{I}_k$	Index set for samples with class label k
$\mathcal{I}_g$	Index set for samples with sensitive attribute g
N, n	Number of training samples, batch size
G	Number of sensitive groups
K	Number of classes
m	Embedding dimensionality
d_f	Dimensionality of Fair Space ($= \|\mathbf{b}\|_0$)
Model Components	
f_ϕ	Encoder $f_\phi : \mathcal{X} \to \mathcal{Z}$
g_ψ	Classifier $g_\psi : \mathcal{Z} \to \mathbb{R}^K$
h_θ	Full model $h_\theta = g_\psi \circ (\mathbf{b} \odot f_\phi)$
$\mathbf{b}$	Binary mask $\in \{0, 1\}^m$
$\mathbf{z}$	Embedding $= f_\phi(\mathbf{x})$
$\mathbf{z}_f$	Fair Space $= \mathbf{b} \odot \mathbf{z}$
$\mathbf{z}_s$	Sensitive Space $= (\mathbf{1} - \mathbf{b}) \odot \mathbf{z}$
Batch Notation	
$\mathbf{Z}$	Batch of embeddings $\in \mathbb{R}^{n \times m}$
$\mathbf{Z}_f, \mathbf{Z}_s$	Batch in the Fair Space and in the Sensitive Space
$\mathbf{S}$	Batch of one-hot sensitive attributes $\in \{0, 1\}^{n \times G}$
Centroids and Means	
$\boldsymbol{\mu}_k$	Class centroid in full $\mathcal{Z}$ (before masking)
$\boldsymbol{\nu}_g$	Group centroid in full $\mathcal{Z}$ (before masking)
$\boldsymbol{\xi}_g, \bar{\boldsymbol{\xi}}$	Group and overall mean output (proofs)
Loss and Hyperparameters	
$\mathcal{L}_{\text{task}}, \mathcal{L}_{\text{comp}}, \mathcal{L}_{\text{reg}}, \mathcal{L}_{\text{total}}$	Task, compression, regularization, and total losses
λ_f, λ_s	Compression loss weights
ρ_f, ρ_s	HSIC regularization weights
σ	Bandwidth of the Gaussian kernel
Mask	
$V_j^{(y)}$	Within-class variance along dim j
$V_j^{(s)}$	Within-group variance along dim j
b_j^*	Closed-form mask value for dim j
α	Mask moving-average rate
Fairness and Theory	
Δ_{DP}	Demographic parity gap of the logits (theory)
p_g	Prevalence of group g: $\mathbb{P}(s = g)$
$p_{\min} = p_{(1)} \leq p_{(2)}$	Smallest and second-smallest group prevalences
$\Delta\pi$	Gap between the groups' positive base rates
c_G	Group constant of the DP bound (Equation 16)
$\mathbf{W}, B$	Weight matrix of the linear head and bound on its entries

B Technical Preliminaries

Here, we provide the definitions regarding the Hilbert-Schmidt Independence Criterion (HSIC) that are used in our method and in the subsequent proofs.

B.1 Hilbert-Schmidt Independence Criterion (HSIC)

The Hilbert-Schmidt Independence Criterion [20] is a kernel-based measure of statistical dependence.

Definition B.1 (HSIC)**.** Let $\mathcal{H}_u$ and $\mathcal{H}_v$ be reproducing kernel Hilbert spaces (RKHS) of functions on domains $\mathcal{U}$ and $\mathcal{V}$, with kernels k_u and k_v. For random variables $U \in \mathcal{U}$ and $V \in \mathcal{V}$, the HSIC is defined as

$$\mathrm{HSIC}(U, V) := \|C_{UV}\|^2_{\mathrm{HS}}, \tag{10}$$

where $C_{UV} = \mathbb{E}\big[(k_u(U,\cdot) - \mathbb{E}k_u(U,\cdot)) \otimes (k_v(V,\cdot) - \mathbb{E}k_v(V,\cdot))\big]$ is the cross-covariance operator and $\|\cdot\|_{\mathrm{HS}}$ denotes the Hilbert-Schmidt norm.

If both kernels are characteristic (e.g., Gaussian RBF), $\mathrm{HSIC}(U, V) = 0$ if and only if $U \perp V$.

Empirical Estimation. Given n i.i.d. samples $\{(u_i, v_i)\}_{i=1}^n$, HSIC can be estimated empirically [20] as:

$$\widehat{\mathrm{HSIC}}(U, V) = (n-1)^{-2}\,\mathrm{tr}(\mathbf{K}_u\mathbf{H}\mathbf{K}_v\mathbf{H}), \tag{11}$$

where $\mathbf{K}_u, \mathbf{K}_v \in \mathbb{R}^{n\times n}$ are kernel matrices with elements $(\mathbf{K}_u)_{ij} = k_u(u_i, u_j)$ and $(\mathbf{K}_v)_{ij} = k_v(v_i, v_j)$, and $\mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ is the centering matrix. On the embeddings we use a Gaussian kernel $k_u(\mathbf{u}, \mathbf{u}') = \exp\big(-\|\mathbf{u} - \mathbf{u}'\|_2^2/(2\sigma^2 q)\big)$ for $\mathbf{u} \in \mathbb{R}^q$, with bandwidth σ, and on the one-hot sensitive attribute $\mathbf{s}$ a linear kernel. In practice, following prior work [41, 52], we use the normalized cross-covariance operator (NOCCO) [17] to obtain a normalized HSIC penalty.

Definition B.2 (HSIC with Linear Kernels)**.** Let $\mathbf{u} \in \mathbb{R}^q$ be a random vector and let $\mathbf{s}$ be the one-hot encoding of s. With linear kernels on both spaces, HSIC reduces to:

$$\mathrm{HSIC}(\mathbf{u}, \mathbf{s}) := \|\mathbf{C}\|_F^2 = \sum_{g=1}^{G} \|\mathrm{Cov}(\mathbf{u}, s_g)\|_2^2, \tag{12}$$

where $\mathbf{C} = \mathrm{Cov}(\mathbf{u}, \mathbf{s}) \in \mathbb{R}^{q\times G}$ is the cross-covariance matrix, and $s_g = \mathbb{1}[s = g]$ is the g-th component of $\mathbf{s}$. Throughout our theoretical analysis, $\mathrm{HSIC}(\cdot,\cdot)$ denotes this instantiation.

C Proof of Proposition 3.1 (Closed-Form Mask)

Proof. Since the mask applies element-wise, the two terms of $\mathcal{L}_{\text{comp}}$ with equal weights decompose across dimensions. For each $j \in \{1, \ldots, m\}$, we solve:

$$\min_{b_j \in [0,1]} \; b_j^2 V_j^{(y)} + (1 - b_j)^2 V_j^{(s)}, \tag{13}$$

where $V_j^{(y)} = \sum_{k=1}^{K} \sum_{i\in\mathcal{I}_k} (z_{ij} - \boldsymbol{\mu}_{kj})^2$ and $V_j^{(s)} = \sum_{g=1}^{G} \sum_{i\in\mathcal{I}_g} (z_{ij} - \boldsymbol{\nu}_{gj})^2$ are the within-class and within-group variances of dimension j. The first measures how much dimension j varies among samples of the same class, the second how much it varies among samples of the same sensitive group. Let $J(b_j)$ denote this objective. Taking the derivative:

$$\frac{\partial J}{\partial b_j} = 2b_j V_j^{(y)} - 2(1 - b_j)V_j^{(s)}. \tag{14}$$

Setting to zero and solving for b_j:

$$b_j V_j^{(y)} = (1 - b_j)V_j^{(s)} \quad \Rightarrow \quad b_j^* = \frac{V_j^{(s)}}{V_j^{(y)} + V_j^{(s)}}. \tag{15}$$

Since $V_j^{(y)}, V_j^{(s)} \geq 0$ are not both zero, the objective is strictly convex, so b_j^* is its unique minimizer, and since $b_j^* \in [0, 1]$, it is also the minimizer over $[0, 1]$. Thus b_j^* is close to 1, assigning dimension j to the Fair Space, when $V_j^{(y)}$ is small relative to $V_j^{(s)}$. □

D Proof of Theorem 3.2 (Main Result)

In this section, we prove the demographic parity bound (Theorem 3.2). The proof follows two steps:

1. We first connect the demographic parity gap Δ_{DP} of an arbitrary predictor to the HSIC between its outputs and the sensitive attribute (Lemma D.1).
2. We then decompose this dependence through the masked linear classification head to conclude the proof of Theorem 3.2.

D.1 Connecting the Demographic Parity to HSIC

Lemma D.1 (Multi-Group Covariance-Demographic Parity Bound)**.** *For a categorical sensitive attribute $s \in \{1, \dots, G\}$ with group prevalences $p_g = \mathbb{P}(s = g)$, the two smallest of which are $p_{(1)} \le p_{(2)}$ with $p_{(1)} > 0$, and any measurable function $h : \mathcal{X} \to \mathbb{R}^K$ with $\mathbb{E}[\|h(\mathbf{x})\|_2] < \infty$:*

$$\Delta_{DP}(h) \le c_G \sqrt{\text{HSIC}(h(\mathbf{x}), \mathbf{s})}, \qquad c_G = \sqrt{\frac{(G-1)\big(p_{(1)}^{-2} + p_{(2)}^{-2}\big) + 2\, p_{(1)}^{-1} p_{(2)}^{-1}}{G}}, \tag{16}$$

where $\mathbf{s}$ *is the one-hot encoding of* s*.*

Proof. Let $\boldsymbol{\xi}_g := \mathbb{E}[h(\mathbf{x}) \mid s = g]$ and $\bar{\boldsymbol{\xi}} := \mathbb{E}[h(\mathbf{x})] = \sum_{g=1}^{G} p_g \boldsymbol{\xi}_g$.

Step 1. Compute covariance with each group indicator. For each indicator $s_g = \mathbb{1}[s = g]$:

$$\text{Cov}(h(\mathbf{x}), s_g) = \mathbb{E}[h(\mathbf{x}) \cdot s_g] - \mathbb{E}[h(\mathbf{x})] \cdot \mathbb{E}[s_g] = p_g \boldsymbol{\xi}_g - p_g \bar{\boldsymbol{\xi}} = p_g(\boldsymbol{\xi}_g - \bar{\boldsymbol{\xi}}). \tag{17}$$

Step 2. Express HSIC. Under linear kernels (Definition B.2):

$$\text{HSIC}(h(\mathbf{x}), \mathbf{s}) = \sum_{g=1}^{G} \|\text{Cov}(h(\mathbf{x}), s_g)\|_2^2 = \sum_{g=1}^{G} p_g^2 \|\boldsymbol{\xi}_g - \bar{\boldsymbol{\xi}}\|_2^2. \tag{18}$$

Step 3. Bound the gap between two groups. Fix two groups $g \neq g'$ and write $t = p_g \|\boldsymbol{\xi}_g - \bar{\boldsymbol{\xi}}\|_2$ and $t' = p_{g'} \|\boldsymbol{\xi}_{g'} - \bar{\boldsymbol{\xi}}\|_2$, so that t^2 and t'^2 are their terms in the sum of Step 2. By the triangle inequality,

$$\|\boldsymbol{\xi}_g - \boldsymbol{\xi}_{g'}\|_2 \le \|\boldsymbol{\xi}_g - \bar{\boldsymbol{\xi}}\|_2 + \|\boldsymbol{\xi}_{g'} - \bar{\boldsymbol{\xi}}\|_2 = \frac{t}{p_g} + \frac{t'}{p_{g'}}. \tag{19}$$

Step 4. Relate the two groups to the remaining groups. Since $\bar{\boldsymbol{\xi}}$ is the weighted average of the group means, the weighted deviations $p_h(\boldsymbol{\xi}_h - \bar{\boldsymbol{\xi}})$ sum to zero over all groups. If two groups deviate from $\bar{\boldsymbol{\xi}}$ in the same direction, the other groups must deviate in the opposite direction. Their deviations are also terms of the sum in Step 2, so they increase HSIC. For $G = 2$, this forces $t = t'$, so $\text{HSIC}(h(\mathbf{x}), \mathbf{s}) = 2t^2$. For $G \ge 3$, the other $G - 2$ groups must together offset the two, and by the Cauchy-Schwarz inequality and the reverse triangle inequality, $\|p_g(\boldsymbol{\xi}_g - \bar{\boldsymbol{\xi}}) + p_{g'}(\boldsymbol{\xi}_{g'} - \bar{\boldsymbol{\xi}})\|_2 \ge |t - t'|$, their terms in Step 2 add at least $(t - t')^2/(G - 2)$, so

$$\text{HSIC}(h(\mathbf{x}), \mathbf{s}) \ge t^2 + t'^2 + \frac{(t - t')^2}{G - 2}. \tag{20}$$

Step 5. Bound the demographic parity gap. By Step 3, the gap between groups g and g' is at most $t/p_g + t'/p_{g'}$, and by Step 4, t and t' cannot both be large for a given HSIC. For $G \ge 3$, Step 4 constrains (t, t') by a quadratic form, and maximizing the linear function $t/p_g + t'/p_{g'}$ under it is a direct application of the Cauchy-Schwarz inequality, using the inverse $\frac{1}{G}\left(\begin{smallmatrix} G-1 & 1 \\ 1 & G-1 \end{smallmatrix}\right)$ of the matrix of that form. For $G = 2$, Step 4 gives $t = t'$ directly. The largest possible value of $t/p_g + t'/p_{g'}$ is therefore reached when t and t' are proportional to

$$\frac{G-1}{p_g} + \frac{1}{p_{g'}} \quad \text{and} \quad \frac{1}{p_g} + \frac{G-1}{p_{g'}}, \tag{21}$$

and equals $c_G \sqrt{\text{HSIC}(h(\mathbf{x}), \mathbf{s})}$ when g, g' are the two rarest groups. Any other pair of groups gives no larger value, so

$$\Delta_{\text{DP}}(h) \le c_G \sqrt{\text{HSIC}(h(\mathbf{x}), \mathbf{s})}. \tag{22}$$

Finally, $c_G \le \sqrt{2}/p_{(1)}$, since $2p_{(1)}^{-1}p_{(2)}^{-1} \le p_{(1)}^{-2} + p_{(2)}^{-2}$ and $p_{(2)}^{-1} \le p_{(1)}^{-1}$. □

D.2 Proof of the Main Theorem

Proof of Theorem 3.2. Throughout, the model computes $h_\theta(\mathbf{x}) = \mathbf{W}\mathbf{z}_f$, where $\mathbf{z}_f = \mathbf{b} \odot f_\phi(\mathbf{x})$ is the Fair Space and $\mathbf{W} \in \mathbb{R}^{K\times m}$ is the weight matrix of the linear classification head.

Step 1. Apply Lemma D.1. Since $\|\mathbf{W}\mathbf{z}_f\|_2 \le \|\mathbf{W}\|_F \|\mathbf{z}_f\|_2$, we have $\mathbb{E}[\|h_\theta(\mathbf{x})\|_2] \le \|\mathbf{W}\|_F \mathbb{E}[\|\mathbf{z}_f\|_2] < \infty$, so Lemma D.1 applies to h_θ:

$$\Delta_{\mathrm{DP}}(h_\theta) \le c_G \sqrt{\mathrm{HSIC}(h_\theta(\mathbf{x}), \mathbf{s})}. \tag{23}$$

Step 2. Decompose through the linear classification head. Since $h_\theta(\mathbf{x}) = \mathbf{W}\mathbf{z}_f$ is linear in $\mathbf{z}_f$:

$$\mathrm{Cov}(h_\theta(\mathbf{x}), s_g) = \mathbf{W}\,\mathrm{Cov}(\mathbf{z}_f, s_g). \tag{24}$$

Let $\mathcal{A} = \{j : b_j = 1\}$ with $|\mathcal{A}| = d_f$, let $\mathbf{z}_\mathcal{A}$ denote $\mathbf{z}_f$ restricted to the active dimensions, and let $\mathbf{W}_\mathcal{A} \in \mathbb{R}^{K\times d_f}$ be the submatrix of $\mathbf{W}$ with the columns in $\mathcal{A}$. Since the inactive dimensions of $\mathbf{z}_f$ are identically zero, their covariances with s_g vanish, so $\mathbf{W}\,\mathrm{Cov}(\mathbf{z}_f, s_g) = \mathbf{W}_\mathcal{A}\,\mathrm{Cov}(\mathbf{z}_\mathcal{A}, s_g)$. Collecting these columns into $\mathbf{C}_\mathcal{A} \in \mathbb{R}^{d_f\times G}$ and summing over g (Definition B.2):

$$\mathrm{HSIC}(h_\theta(\mathbf{x}), \mathbf{s}) = \|\mathbf{W}_\mathcal{A}\mathbf{C}_\mathcal{A}\|_F^2. \tag{25}$$

Step 3. Apply norm bounds. Using the submultiplicativity of the Frobenius norm[2]:

$$\|\mathbf{W}_\mathcal{A}\mathbf{C}_\mathcal{A}\|_F \le \|\mathbf{W}_\mathcal{A}\|_F \cdot \|\mathbf{C}_\mathcal{A}\|_F. \tag{26}$$

We bound the two terms separately:

1. **Weight Matrix:** Since $|W_{ij}| \le B$ for all entries (as $\|\mathbf{W}\|_{\max} \le B$) and $\mathbf{W}_\mathcal{A}$ has dimensions $K \times d_f$:

$$\|\mathbf{W}_\mathcal{A}\|_F = \sqrt{\sum_{i=1}^{K}\sum_{j\in\mathcal{A}} W_{ij}^2} \le \sqrt{K \cdot d_f \cdot B^2} = B\sqrt{K d_f}. \tag{27}$$

2. **Covariance Matrix:** Since the inactive dimensions of $\mathbf{z}_f$ contribute zero covariance, $\|\mathbf{C}_\mathcal{A}\|_F^2$ equals the full sum of squared covariances in Definition B.2. Therefore:

$$\|\mathbf{C}_\mathcal{A}\|_F = \sqrt{\mathrm{HSIC}(\mathbf{z}_f, \mathbf{s})}. \tag{28}$$

Combining these bounds with Step 2 yields:

$$\sqrt{\mathrm{HSIC}(h_\theta(\mathbf{x}), \mathbf{s})} \le B\sqrt{K d_f} \cdot \sqrt{\mathrm{HSIC}(\mathbf{z}_f, \mathbf{s})}. \tag{29}$$

Substituting this into the inequality from Step 1 completes the proof. □

[2]The inequality $\|\mathbf{A}\mathbf{B}\|_F \le \|\mathbf{A}\|_F\|\mathbf{B}\|_F$ follows from applying Cauchy-Schwarz to each entry of the product.

Table 7: Dataset statistics. K and G denote the number of target classes and sensitive groups, respectively. Target Dist. shows the class distribution, and Sens. Dist. shows the sensitive group distribution. FairFace is used in Section 4.3, where each seed samples $60{,}000$ training and $10{,}000$ test images. Groups are ordered Female/Male (Adult, CelebA-A), White/Black/Other (UTKFace-AR), Male/Female (FairFace), and Male-Older/Male-Young/Female-Older/Female-Young (COMPAS).

NAME	MODALITY	TARGET	SAMPLES	K	G	TARGET DIST. (%)	SENS. DIST. (%)
ADULT	TABULAR	INCOME	45,222	2	2	75.2 / 24.8	32.5 / 67.5
COMPAS	TABULAR	2-YEAR RECIDIVISM	5,278	2	4	53.0 / 47.0	58.8 / 21.7 / 14.6 / 5.0
CELEBA-A	IMAGE	ATTRACTIVE	202,599	2	2	48.7 / 51.3	58.3 / 41.7
UTKFACE-AR	IMAGE	AGE (3-CLASS)	23,705	3	3	32.2 / 44.1 / 23.7	42.5 / 19.1 / 38.4
FAIRFACE	IMAGE	AGE ≥ 30	97,698	2	2	54.2 / 45.8	53.0 / 47.0

E Algorithm Pseudocode

Algorithm 1 gives the full pseudocode of FAIRSPLIT (Section 3). For each minibatch, it alternates two steps: (i) a gradient step on the model parameters θ, with moving-average updates of the centroids, and (ii) a closed-form update of the mask (Proposition 3.1).

Algorithm 1 FAIRSPLIT

Input: Dataset $\mathcal{D}$, hyperparameters $\lambda_f, \lambda_s, \rho_f, \rho_s$, moving-average rate α, learning rate η, number of iterations T
Initialize: Model parameters $\theta^{(0)}$, global centroids $\boldsymbol{\mu}$ (class) and $\boldsymbol{\nu}$ (group), mask $\mathbf{b}^{(0)}$ via Equation 7, binarized at 0.5
for iteration $t = 1$ to T **do**
    Sample minibatch $\mathcal{B} = \{(\mathbf{x}_i, y_i, s_i)\} \subset \mathcal{D}$
    // — Step 1: Representation update —
    Compute embeddings $\mathbf{z}_i = f_\phi(\mathbf{x}_i)$ for $i \in \mathcal{B}$
    Compute batch means: $\tilde{\boldsymbol{\mu}}_k$ (class), $\tilde{\boldsymbol{\nu}}_g$ (group)
    Update global centroids $\boldsymbol{\mu}_k, \boldsymbol{\nu}_g$ with batch means
    Compute loss $\mathcal{L}_{\text{total}}(\mathcal{B}; \theta, \mathbf{b})$
    Update $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{total}}$
    // — Step 2: Mask update (fix θ) —
    Detach the embeddings $\mathbf{z}_i$ (stop gradients)
    for dimension $j = 1$ to m **do**
        $V_j^{(y)} \leftarrow \sum_k \sum_{i \in \mathcal{I}_k} (z_{ij} - \boldsymbol{\mu}_{kj})^2$
        $V_j^{(s)} \leftarrow \sum_g \sum_{i \in \mathcal{I}_g} (z_{ij} - \boldsymbol{\nu}_{gj})^2$
        Compute closed-form mask value:
        $b_j^* \leftarrow \frac{V_j^{(s)}}{V_j^{(y)} + V_j^{(s)}}$
        Update mask:
        Smooth b_j^* with a moving average of rate α and binarize it at 0.5 to obtain b_j
    end for
end for
Return: θ, $\mathbf{b}$

F Reproducibility and Experimental Details

F.1 Datasets

We evaluate on five common fairness datasets: Adult and COMPAS as tabular settings, CelebA-A and UTKFace-AR as image settings, and FairFace for the ViT-B/16 experiment. Table 7 summarizes all five datasets. The COMPAS sensitive attribute is intersectional (gender $\times$ age), yielding four groups with $p_{\min} = 0.05$, and UTKFace-AR uses 3-class age prediction with 3-group race as the sensitive attribute. For the experiments with large pretrained models (Section 4.3), we use FairFace at 224×224 with a fine-tuned ViT-B/16, and CelebA at 224×224 with frozen foundation models.

Adult [35, 4] is a tabular dataset where the task is to predict whether an individual's annual income exceeds \$50K. The sensitive attribute is gender (Female/Male). The dataset is hosted by the UCI Machine Learning Repository[3] and is distributed under the Creative Commons Attribution 4.0 International (CC BY 4.0) license.

COMPAS [2] is a tabular dataset where the task is to predict whether a defendant will reoffend within two years (binary recidivism). We apply the standard ProPublica preprocessing and restrict the data to African-American and Caucasian defendants. The sensitive attribute is intersectional, combining gender (Female/Male) with a binarized age group (Young: ≤ 25, Older: > 25), yielding four groups with minimum prevalence $p_{\min} = 0.05$. The data is released by ProPublica[4] under the ProPublica Data Store terms of use.

CelebA-A [39] consists of celebrity face images where we predict the "Attractive" attribute. The sensitive attribute is gender (Female/Male). Images are resized to 48×48 RGB. For the frozen foundation-model experiments (Appendix G.3), we instead use CelebA at 224×224 resolution, learn the Fair Space for predicting high cheekbones, and transfer it to predicting age (the `Young` attribute), with gender again as the sensitive attribute. We use the official split and hold out 10% of the training images for validation ($146{,}493$ / $16{,}277$ / $19{,}962$ train / validation / test images). For CLIP ViT-L/14 and CLIP ResNet-50 [46], we use the precomputed image features released by Dehdashtian et al. [10], and for DINOv2 ViT-L/14 [43] we extract features from the aligned images. Per the agreement on the official project page[5], the dataset is available for non-commercial research purposes only.

UTKFace-AR [56] contains face images with age and ethnicity annotations. We bin age into three groups: Young (≤ 25), Middle-aged (26-45), and Old (> 45), which yields reasonably balanced classes (32%, 44%, 24%). The sensitive attribute is race, grouped into three categories: White, Black, and Other (Asian, Indian, and Others combined). Images are 48×48 grayscale, replicated to three channels. Per the terms on the official project page[6], the dataset is available for non-commercial research purposes only.

FairFace [32] is a face image dataset with a balanced race composition. We use it for the end-to-end ViT-B/16 experiment (Appendix G.3) at its native 224×224 resolution, predicting whether age is at least 30, with gender as the sensitive attribute. For each seed, we sample $60{,}000$ training images from the official training split and $10{,}000$ test images from the official validation split. The dataset is released under the CC BY 4.0 license.

F.2 Metrics

Table 8 summarizes the evaluation metrics used in our experiments. All values are reported in percent. The hard prediction $\hat{y}$ is obtained by thresholding the sigmoid output at 0.5 for binary classification and by taking the argmax of the softmax output for multiclass. The threshold-free metrics (AP, ABCC, MCDP) operate directly on predicted probabilities. For datasets with a binary target and more than two groups (COMPAS), we use the variants from the right column with the k index dropped, so each fairness metric becomes the maximum disparity across all pairs of sensitive groups.

F.3 Implementation

Our implementation builds upon the code of Han et al. [22][7] and Jin et al. [28][8], and adapts the mask and HSIC components from H-SPLID [41][9]. We port MMD-B-Fair [11] from its original repository[10] into the FFB benchmark setup. Architecture details (MLP encoder for tabular settings, ResNet-18 backbone for image settings) and the linear classification head are described in Section F.5.

[3] `https://archive.ics.uci.edu/dataset/2/adult`

[4] Source: `https://github.com/propublica/compas-analysis`. Terms: `https://projects.propublica.org/datastore/terms/`.

[5] `https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html`

[6] `https://susanqq.github.io/UTKFace/`

[7] `https://github.com/ahxt/fair_fairness_benchmark`

[8] `https://github.com/mitao-cat/icml24_mcdp`

[9] `https://github.com/neu-spiral/H-SPLID`

[10] `https://github.com/namratadeka/mmd-b-fair`

Table 8: Performance and fairness metrics. $\hat{y} \in \{1, \ldots, K\}$ is the predicted class, y the true class, and $s \in \{1, \ldots, G\}$ the sensitive attribute, with class 2 as the positive class for binary targets. P and R are precision and recall, and P_r, R_r their values at the r-th threshold. With $\mathbb{P}_g(\cdot) = \mathbb{P}(\cdot \mid s = g)$, $\mathrm{TPR}_g^{(k)} = \mathbb{P}_g(\hat{y}{=}k \mid y{=}k)$, $\mathrm{FPR}_g^{(k)} = \mathbb{P}_g(\hat{y}{=}k \mid y{\neq}k)$, and $F_g^{(k)}$ the CDF of the predicted probability of class k in group g, $\Delta(\cdot)_{g,g'}$ denotes the difference between groups g and g'. Binary metrics drop the indices.

Metric	Name	Binary	Multiclass / Multi-group
AP ↑	Average Precision	$\sum_r (R_r - R_{r-1}) P_r$	–
F1 ↑	F1 Score	$2PR/(P+R)$	$\frac{1}{K}\sum_k 2P_kR_k/(P_k+R_k)$
ΔDP ↓	Demographic Parity	$\lvert\Delta\mathbb{P}(\hat{y}{=}2)\rvert$	$\max_{k,g,g'} \lvert\Delta\mathbb{P}_{g,g'}(\hat{y}{=}k)\rvert$
ΔEOpp ↓	Equal Opportunity	$\lvert\Delta\mathrm{TPR}\rvert$	$\max_{k,g,g'} \lvert\Delta\mathrm{TPR}^{(k)}_{g,g'}\rvert$
ΔEOdd ↓	Equalized Odds	$\lvert\Delta\mathrm{TPR}\rvert + \lvert\Delta\mathrm{FPR}\rvert$	$\max_{k,g,g'} \left(\lvert\Delta\mathrm{TPR}^{(k)}_{g,g'}\rvert + \lvert\Delta\mathrm{FPR}^{(k)}_{g,g'}\rvert\right)$
ABCC ↓	Area Between CDF Curves	$\int_0^1 \lvert\Delta F(t)\rvert \, \mathrm{d}t$	$\max_{k,g,g'} \int_0^1 \lvert\Delta F^{(k)}_{g,g'}(t)\rvert \, \mathrm{d}t$
MCDP ↓	Maximum CDF Disparity	$\max_t \lvert\Delta F(t)\rvert$	$\max_{k,g,g',t} \lvert\Delta F^{(k)}_{g,g'}(t)\rvert$

F.4 Software and Hardware Setup

Our implementation is written in Python using the PyTorch framework. The source code is publicly available at `https://github.com/shkabrii/fairsplit`. All experiments were conducted on a single internal server running Ubuntu 22.04.3 LTS. The server is equipped with 64 CPUs, 512 GiB of system memory, and a single NVIDIA A100 GPU with 80 GB of VRAM.

F.5 Hyperparameters

For binary classification with binary sensitive attributes (Adult, CelebA-A), we follow the in-processing fair classification benchmark of Han et al. [22]. For non-binary sensitive attributes (UTKFace-AR, COMPAS) and the multiclass target of UTKFace-AR, we make the following extensions. Baselines restricted to binary targets and sensitive attributes are excluded, namely FairMixup, DRAlign, DiffMCDP, PRemover, and MMD-B-Fair (see Table 1). For UTKFace-AR, performance is reported as F1 rather than AP, and val F1 replaces val AP in the $\geq 95\%$-of-ERM gate of the selection rule. Fairness metrics are extended to multiple classes and groups by taking the maximum disparity across all classes and all pairs of sensitive groups (formal definitions in Table 8).

For all experiments, we use the Adam optimizer [33]. In tabular experiments, we employ an MLP encoder with hidden layers $[512, 256, 64]$. For image experiments, we use a ResNet-18 backbone pretrained on ImageNet, where the final fully connected layer is modified to match the target dimensionality. Images are resized to 48×48, following the FFB benchmark [22], and pixel values are scaled to $[0, 1]$. For the frozen backbones, the encoder is a projection on top of the precomputed features, an MLP with hidden layers $[1024, 512, 256]$ and no dropout, so the embedding has $m{=}256$ dimensions.

On Adult, COMPAS, CelebA-A, and UTKFace-AR, we use a learning rate of 1×10^{-3} and a `StepLR` scheduler with step size 50 and decay factor $\gamma = 0.1$. Training is conducted for 150 steps in total, following the FFB protocol [22]. In our preliminary experiments, we observed that the fairness regularizers worked better when added to a classifier that had already learned the task than when applied from the start. We therefore train every fairness method with the classification loss only for the first 50 steps, after which its fairness objective is added and the optimizer and its learning-rate schedule are reset. ERM is trained without this warm-up. The 224×224 experiments (Appendix G.3) use longer schedules. The ViT-B/16 is fine-tuned for 300 steps with a learning rate of 10^{-4}, decayed every 100 steps, and a batch size of 128. The projection on top of the frozen backbones is trained for 600 steps with a learning rate of 3×10^{-4}, decayed every 200 steps, and a batch size of 1024. In both cases, FAIRSPLIT also starts with 50 steps of the classification loss only.

For each fairness method, we run a grid search tuning all of its method-specific hyperparameters, with per-dataset settings differing only in encoder, batch size, and target and sensitive attributes. The selected configuration per (method, dataset) is in Table 10 for baselines and in Table 11 for FAIRSPLIT.

Table 9: Training hyperparameters shared by all methods. The mask moving-average rate applies to FAIRSPLIT only.

Hyperparameter	**Benchmarks**	**ViT-B/16**	**Frozen backbones**
Optimizer	Adam	Adam	Adam
Learning rate	10^{-3}	10^{-4}	3×10^{-4}
StepLR step size (γ=0.1)	50	100	200
Training steps	150	300	600
Warm-up steps (fairness methods)	50	50	50
Batch size	1024 (tabular), 128 (image)	128	1024
Mask moving-average rate α	0.99	0.99	0.99

Table 10: Selected best hyperparameters per baseline method and dataset. Among the configurations whose validation AP (validation F1 for UTKFace-AR) reaches at least 95% of ERM's, we select the one with the lowest mean validation ΔDP (see Section 4). Cells marked "—" indicate the method is not applicable to the dataset's setting.

Method	**Parameter**	**Adult**	**CelebA-A**	**UTKFace-AR**	**COMPAS**
AdvDebias	λ	2	0.2	0.35	1
FairMixup	λ	0.8	1	—	—
DRAlign	λ	0.5	0.2	—	—
	β	0.01	0.01	—	—
HSIC(Y, S)	λ	70	50	300	200
	σ	1	5	5	1
DiffMCDP	λ	0.5	0.3	—	—
	τ	5	20	—	—
PRemover	λ	0.005	0.025	—	—
MMD-B-Fair	λ	0.5	0.2	—	—
	σ range	$(0.5, 2.0)$	$(0.7, 3.0)$	—	—

G Additional Results

G.1 Bound Tightness

We evaluate the bound of Theorem 3.2 on trained models. For each run, we compute the right-hand side of Equation 9 on the test set, with B the largest absolute entry of the trained classification head and HSIC the linear-kernel HSIC of Definition B.2, and compare it with the demographic parity gap $\Delta_{\mathrm{DP}}(h_\theta)$ of the logits. Figure 5 reports both on Adult, CelebA-A, UTKFace-AR, and COMPAS as ρ_f varies, together with the demographic parity gap of the predicted classes. The bound follows the gap as ρ_f changes. At the selected ρ_f, it exceeds the gap by a factor of 2.0 on Adult, 2.4 on CelebA-A, 5.5 on UTKFace-AR, and 14.3 on COMPAS. The factor is largest on COMPAS, whose smallest group has a prevalence of 0.05, since c_G grows with $1/p_{\min}$.

G.2 Robustness Across Encoders and Embedding Dimensionalities

We evaluate FAIRSPLIT across five pretrained encoders spanning embedding dimensionalities $m \in \{512, 960, 1280, 2048\}$. FAIRSPLIT uses the same hyperparameters for every encoder, and only the encoder is changed, with no per-architecture retuning. This setup tests two properties at once: (i) whether FAIRSPLIT is robust to the choice of encoder, and (ii) whether the closed-form mask (Proposition 3.1) adapts d_f to embedding spaces of different sizes. Results on UTKFace-AR and CelebA-A are reported in Table 12, averaged over 5 seeds. We report F1↑ (UTKFace-AR) or AP↑ (CelebA-A) as the utility metric, ΔDP↓ as the fairness metric, the relative ΔDP reduction over ERM, and the learned Fair Space dimensionality d_f together with the encoder's embedding dimensionality m.

Across all five backbones, FAIRSPLIT reduces the demographic parity gap by between 45% and 56% on UTKFace-AR and between 38% and 65% on CelebA-A, at a utility cost that varies by encoder.

Table 11: Selected FAIRSPLIT hyperparameters per dataset, with the same selection rule as in Table 10. σ is fixed at the modality default (tabular vs. image). For the frozen backbones, the hyperparameters are selected on CLIP ViT-L/14 and reused unchanged on the other two backbones.

Parameter	Adult	CelebA-A	UTKFace-AR	COMPAS	ViT-B/16	Frozen
λ_f	0.025	0.01	0.1	0.01	0.01	0.01
λ_s	0.05	0.025	0.05	0.2	0.025	0.05
ρ_f	0.4	0.5	1.3	0.3	1.5	1.0
ρ_s	0.005	0.5	0.1	0.005	2.0	0.5
σ	2	5	5	2	5	5

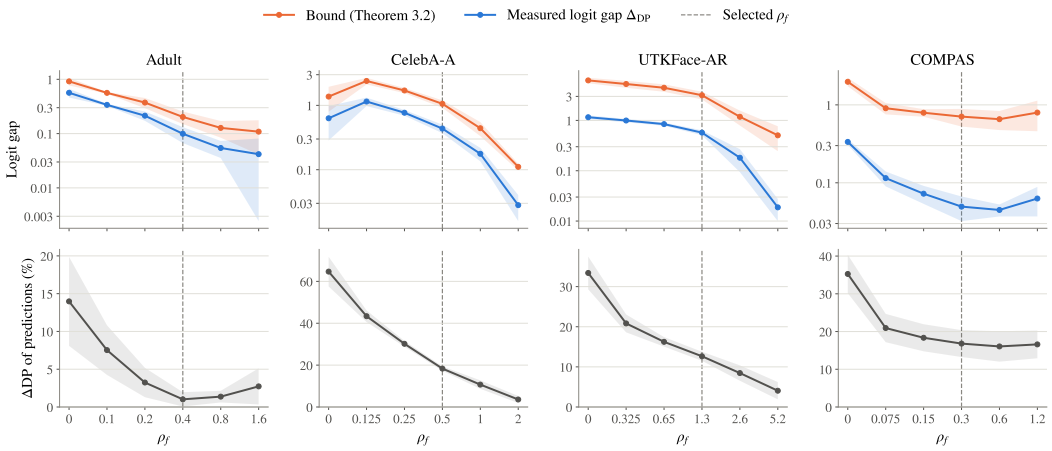


Figure 5: Bound of Theorem 3.2 and the demographic parity gap of the logits (top), and the demographic parity gap of the predicted classes (bottom), as the HSIC regularization strength ρ_f varies. Shaded regions are the mean $\pm$ std over 5 seeds, and the dashed line marks the reported ρ_f.

The mask adapts d_f to each embedding without manual thresholding or per-encoder tuning. The active fraction d_f/m varies across backbones, for example $233/512 \approx 46\%$ for ResNet-18 versus $1161/1280 \approx 91\%$ for EfficientNetV2-S. Since the bound of Theorem 3.2 scales with $\sqrt{d_f}$, the mask balances fairness and utility in both low- and high-dimensional embedding spaces. FAIRSPLIT therefore transfers across these five encoders and their embedding sizes without retuning.

Embedding size. To test whether the embedding size alone affects fairness, we vary the dimensionality m of the encoder's output embedding on Adult (MLP layers $[512, 256, m]$) over $m \in \{32, 64, 128, 256, 512\}$, with 5 seeds per setting (Table 13). Unlike in the main tables, the 95% utility selection rule is not applied, so the utility cost varies across sizes. ERM barely changes across the whole range, from 19.8 to 17.5 in ΔDP with ABCC staying flat, even at $m = 32$, which is smaller than the Fair Space that FAIRSPLIT learns at $m = 64$ ($d_f \approx 47$). The embedding size is therefore not by itself a fairness mechanism. FAIRSPLIT keeps ΔDP between 1.0 and 3.3 at every size with no systematic trend, and reduces ABCC from about 18.7 to between 0.8 and 4.3, at a utility cost of 3.2 to 6.5 AP. The fairness gain therefore comes from the decomposition with HSIC-based regularization rather than from the embedding size.

G.3 Fine-Tuned Vision Transformers and Frozen Foundation Models

ViT-B/16. We fine-tune an ImageNet-pretrained ViT-B/16 end-to-end as the FAIRSPLIT encoder on FairFace at its native 224×224 resolution, predicting whether age is at least 30 with gender as the sensitive attribute. ERM uses the same architecture and optimization. FAIRSPLIT lowers ΔDP from 23.0 to 9.5 and more than halves every other fairness metric while retaining 97% of ERM's AP, and the mask keeps 552 of the 768 dimensions (Table 4).

Frozen foundation models and transfer to a related task. We apply FAIRSPLIT on top of frozen CLIP ViT-L/14, CLIP ResNet-50, and DINOv2 ViT-L/14 features of CelebA, with gender as the

Table 12: Encoder robustness on UTKFace-AR (3-class age, 3 racial groups) and CelebA-A. FAIRSPLIT uses the same hyperparameters for every encoder, and only the encoder is changed, with no per-architecture retuning. Since the hyperparameters are not tuned per encoder, the 95% utility gate of the main tables is not applied here. Utility and ΔDP columns show ERM $\rightarrow$ FAIRSPLIT, and Red. is the relative reduction of ΔDP. Mean over 5 seeds.

		UTKFace-AR				CelebA-A			
Encoder	m	F1↑	ΔDP↓	Red.	d_f	AP↑	ΔDP↓	Red.	d_f
ResNet-18	512	73.8 → 70.4	27.9 → 12.6	55%	233	84.5 → 80.4	52.8 → 18.3	65%	334
ResNet-34	512	73.8 → 69.1	28.9 → 12.8	56%	243	84.5 → 78.1	53.2 → 18.4	65%	297
ResNet-50	2048	74.8 → 72.2	27.9 → 14.9	47%	1003	85.9 → 80.6	52.4 → 25.5	51%	1230
EfficientNetV2-S	1280	73.2 → 73.8	29.1 → 15.9	45%	1161	84.9 → 82.6	50.6 → 31.2	38%	777
MobileNetV4-S	960	70.4 → 67.5	28.5 → 13.9	51%	584	84.0 → 74.7	51.6 → 19.3	63%	419

Table 13: Embedding-size ablation on Adult (5 seeds).

		$m = 32$	$m = 64$	$m = 128$	$m = 256$	$m = 512$
ΔDP ↓	ERM	19.8±0.8	19.4±0.6	18.5±0.5	18.2±0.3	17.5±0.2
	FAIRSPLIT	3.3±1.2	1.0±0.9	3.2±2.6	1.3±0.5	2.5±2.4
AP ↑	ERM	76.8±0.5	76.7±0.5	77.0±0.5	77.3±0.4	77.5±0.4
	FAIRSPLIT	71.5±0.8	72.9±0.9	71.6±1.2	70.8±1.2	74.3±1.0
ABCC ↓	ERM	18.8±0.3	18.7±0.2	18.5±0.2	18.6±0.2	18.5±0.1
	FAIRSPLIT	0.9±0.2	2.4±0.7	1.5±0.3	0.8±0.2	4.3±0.5

sensitive attribute. Since the backbone stays frozen, the split acts on a small trainable projection (1024-512-256) trained to predict high cheekbones (Task A). We then keep the projection and the mask fixed and train only a new linear head on $\mathbf{z}_f$ to predict whether a person is young (Task B), without using the sensitive attribute. The hyperparameters selected on CLIP ViT-L/14 are reused unchanged on the other two backbones. ERM uses the same architecture and optimization and follows the same transfer protocol, with its head reading the full embedding. On Task B, which the Fair Space was never trained or tuned on, FAIRSPLIT improves every fairness metric on all three backbones while retaining at least 95% of ERM's AP (Table 14).

Table 14: Transfer of the Fair Space from Task A (`High_Cheekbones`) to Task B (`Young`) on three frozen backbones (5 seeds). Task B uses a fresh linear head on the frozen representation.

Backbone	Task	Method	AP ↑	ΔDP ↓	ABCC ↓	ΔEOpp ↓	ΔEOdd ↓	d_f/m
CLIP ViT-L/14	A	ERM	**93.7±0.0**	27.8±0.4	26.0±0.2	15.8±0.4	18.3±1.0	256 / 256
		FAIRSPLIT	89.3±0.4	**13.1±1.2**	**9.0±0.6**	**1.2±0.7**	**12.1±1.0**	174 / 256
	B	ERM	**96.5±0.1**	21.4±0.9	20.0±0.6	4.9±0.5	23.6±2.4	256 / 256
		FAIRSPLIT	93.1±0.4	**9.1±1.3**	**6.4±0.7**	**0.8±0.3**	**4.8±1.6**	174 / 256
CLIP ResNet-50	A	ERM	**94.5±0.0**	27.1±0.2	26.3±0.1	14.0±0.5	15.7±0.5	256 / 256
		FAIRSPLIT	90.8±0.7	**15.4±1.3**	**10.5±0.9**	**1.5±1.2**	**10.9±0.6**	168 / 256
	B	ERM	**96.6±0.1**	23.4±0.4	21.0±0.3	6.1±0.5	29.4±1.5	256 / 256
		FAIRSPLIT	93.6±0.4	**12.2±2.4**	**7.9±1.5**	**0.9±0.5**	**8.5±5.0**	168 / 256
DINOv2 ViT-L/14	A	ERM	**94.4±0.0**	28.7±0.2	26.8±0.2	16.8±0.3	19.7±0.5	256 / 256
		FAIRSPLIT	90.7±0.5	**15.2±1.0**	**10.3±0.6**	**1.7±0.8**	**11.2±0.4**	166 / 256
	B	ERM	**95.2±0.1**	23.5±0.4	22.1±0.2	7.3±0.3	35.3±1.7	256 / 256
		FAIRSPLIT	90.9±0.5	**9.1±1.8**	**5.9±1.2**	**0.5±0.4**	**7.1±1.7**	166 / 256

G.4 Probing the Fair and Sensitive Spaces

To measure what a representation reveals about the sensitive attribute, we freeze a trained model and fit probes that predict s from it. The linear probe is a logistic regression, and the MLP probe has two hidden layers of 256 units. Both are fit on training embeddings and evaluated by balanced accuracy

Table 15: Balanced accuracy (%, 5 seeds) of probes predicting the sensitive attribute from the full embedding $\mathbf{z}$, the Fair Space $\mathbf{z}_f$, and the Sensitive Space $\mathbf{z}_s$ of the same trained model, where random guessing scores 50%. CLIP ViT-L/14 is frozen (Appendix G.3), and ResNet-18 is the CelebA-A model of Table 2. For $\mathbf{z}$, AP, and ΔDP, arrows show ERM $\rightarrow$ FAIRSPLIT with the same architecture. Lower probe accuracy means that s is harder to predict.

			Linear probe			MLP probe			
Backbone	Task	$\Delta\pi$	$\mathbf{z}$	$\mathbf{z}_f$	$\mathbf{z}_s$	$\mathbf{z}_f$	d_f/m	AP $\uparrow$	ΔDP $\downarrow$
CLIP ViT-L/14	`Smiling`	14	98.1 $\rightarrow$ 99.0	50.4±0.3	99.0±0.0	95.9±1.1	167 / 256	97.9 $\rightarrow$ 97.0	15.2 $\rightarrow$ 8.5
	`Young`	24	98.5 $\rightarrow$ 99.0	61.0±6.9	99.0±0.0	89.0±1.9	91 / 256	97.6 $\rightarrow$ 95.5	23.2 $\rightarrow$ 4.9
	`High_Cheekbones`	25	98.4 $\rightarrow$ 99.0	54.7±2.1	99.0±0.0	97.2±0.9	174 / 256	93.7 $\rightarrow$ 89.3	27.8 $\rightarrow$ 13.1
	`Wavy_Hair`	31	98.8 $\rightarrow$ 99.0	64.4±4.5	99.0±0.0	94.8±1.1	115 / 256	85.6 $\rightarrow$ 83.3	35.4 $\rightarrow$ 26.6
ResNet-18	`Attractive`	40	85.7 $\rightarrow$ 94.8	88.2±2.4	94.8±0.2	94.5±0.3	334 / 512	84.5 $\rightarrow$ 80.4	52.8 $\rightarrow$ 18.3

Table 16: Balanced accuracy (%, 5 seeds) of probes predicting the sensitive attribute from the input of each method's classifier, the Fair Space $\mathbf{z}_f$ for FAIRSPLIT and the full embedding $\mathbf{z}$ for the baselines. Random guessing scores 50% on Adult, 33% on UTKFace-AR, and 25% on COMPAS. The probed models are those of Tables 2 and 3. Lower probe accuracy means that s is harder to predict, and the lowest value per column is in bold. A dash (—) marks baselines that support only binary targets and binary sensitive attributes, so they cannot be applied to UTKFace-AR (three age classes, three racial groups) or COMPAS (four groups).

	Adult		UTKFace-AR		COMPAS	
Method	Linear	MLP	Linear	MLP	Linear	MLP
ERM	71.7±0.8	79.1±0.9	69.4±0.8	68.2±0.9	45.1±0.9	48.5±0.6
AdvDebias	80.0±2.5	80.1±1.9	70.0±1.2	66.4±1.8	48.7±0.5	47.8±0.7
FairMixup	75.9±0.6	80.6±0.7	—	—	—	—
DRAlign	72.1±2.4	79.7±0.9	—	—	—	—
HSIC(Y, S)	72.0±1.6	78.6±2.5	64.8±1.8	62.2±1.8	44.1±0.7	47.5±0.6
DiffMCDP	73.7±1.4	79.5±2.5	—	—	—	—
PRemover	71.0±2.1	79.3±2.5	—	—	—	—
MMD-B-Fair	68.9±0.7	79.3±1.0	—	—	—	—
FAIRSPLIT	**60.5±1.4**	**78.5±1.6**	**48.9±1.7**	**60.6±1.8**	**25.3±0.3**	**35.2±4.3**

on test embeddings, averaged over 5 seeds. Table 15 compares the full embedding, the Fair Space, and the Sensitive Space of each FAIRSPLIT model. Table 16 compares FAIRSPLIT with the baselines on the input of their classifiers.

Fair and Sensitive Spaces. Table 15 reports four tasks on frozen CLIP ViT-L/14, with the setup of Appendix G.3, and the CelebA-A model of Table 2. In every setting, the Sensitive Space predicts s as well as the full embedding, while a linear probe predicts it less accurately from the Fair Space, at 50-64% on frozen CLIP features. A non-linear MLP probe recovers s from the Fair Space in all settings.

Comparison with baselines. Table 16 probes the Fair Space of FAIRSPLIT and the full embedding of each baseline, which is the input of their classifiers. A linear probe predicts s least accurately from the Fair Space of FAIRSPLIT on Adult, UTKFace-AR, and COMPAS, by 8.4 to 18.8 points below the next-best method, and at the level of random guessing on COMPAS. With the MLP probe, the Fair Space of FAIRSPLIT is also the least predictive on UTKFace-AR and COMPAS, while on Adult all methods lie within 2.1 points.

Scope. These results are consistent with Theorem 3.2, which bounds the demographic parity of a linear head through the linear dependence between $\mathbf{z}_f$ and s. FAIRSPLIT reduces this dependence rather than removing s from the representation entirely. Complete removal would force the group-wise error rates of every classifier to sum to at least $\Delta\pi$ [58].

G.5 Empirical Performance with Non-linear Classifier Heads

Theorem 3.2 requires a linear classification head. We test how much this assumption matters in practice by replacing the linear head with a non-linear MLP head. Table 17 compares the linear-head and non-linear-head settings on CelebA-A and UTKFace-AR.

Table 17: FAIRSPLIT with linear and non-linear classification heads on $\mathbf{z}_f$ (5 seeds). The utility metric is AP on CelebA-A and F1 on UTKFace-AR. For each dataset and head, bold marks the better of ERM and FAIRSPLIT.

Method	Head	AP/F1↑	ΔDP↓	ABCC↓	MCDP↓	ΔEOpp↓	ΔEOdd↓	d_f
CelebA-A								
ERM	Linear	**84.5±0.4**	52.8±0.8	35.9±0.8	53.4±0.8	36.8±1.5	75.0±2.1	512
FAIRSPLIT	Linear	80.4±1.1	**18.3±1.1**	**9.4±1.3**	**19.6±1.8**	**2.6±0.3**	**5.2±1.7**	334
ERM	Non-linear	**85.3±0.4**	53.7±1.9	37.5±1.2	54.0±2.0	37.3±3.0	75.6±4.9	512
FAIRSPLIT	Non-linear	81.7±0.5	**23.1±3.6**	**13.5±1.5**	**24.3±3.3**	**4.4±2.2**	**11.0±5.0**	369
UTKFace-AR								
ERM	Linear	**73.8±0.7**	27.9±1.4	27.1±2.6	36.2±2.4	29.1±5.2	35.9±3.0	512
FAIRSPLIT	Linear	70.4±1.5	**12.6±1.1**	**9.1±1.0**	**18.7±1.2**	**14.9±4.6**	**22.1±4.9**	233
ERM	Non-linear	**74.9±1.0**	30.4±1.7	28.8±1.7	39.9±2.9	33.2±5.4	40.8±4.8	512
FAIRSPLIT	Non-linear	70.3±2.1	**17.5±3.7**	**12.4±1.8**	**21.5±2.6**	**23.4±3.4**	**26.5±2.2**	289

The fairness gains are smaller with the non-linear head, as expected, since a non-linear head can use sensitive information in $\mathbf{z}_f$ that a linear head cannot. They remain substantial: ΔDP drops from 53.7 to 23.1 on CelebA-A (57% reduction) and from 30.4 to 17.5 on UTKFace-AR (42% reduction), with utility within about 5 points of ERM in both settings. The mask keeps $d_f = 369$ dimensions on CelebA-A and 289 on UTKFace-AR. The linear head is therefore needed for the theoretical guarantee, but not for the method to improve fairness in practice.

G.6 Performance under Imposed Group Imbalance

This subsection complements the intersectional COMPAS experiment in the main paper (Table 3) with a controlled imbalance setting on a high-dimensional image dataset. We resample CelebA-A so that one sensitive group constitutes only 5% of the training data. Each method keeps its hyperparameters from Tables 10 and 11, and only the training distribution changes.

Table 18 shows that FAIRSPLIT achieves the best score on all five fairness metrics while matching HSIC(Y, S) on AP. AdvDebias stays close to ERM on every fairness metric under this imbalance.

Table 18: CelebA-A with an imposed $95/5$ group imbalance in the training data (5 seeds). Each method keeps its hyperparameters from Tables 10 and 11. Best in **bold**, second-best underlined.

	ΔDP↓	ABCC↓	MCDP↓	ΔEOpp↓	ΔEOdd↓	AP↑
ERM	54.0±0.4	34.2±0.8	54.4±0.5	42.4±1.8	79.3±1.6	84.1±0.3
AdvDebias	52.3±2.7	35.1±1.1	53.6±1.9	35.4±8.8	73.3±7.0	**85.4±0.3**
HSIC(Y, S)	19.4±1.9	10.2±1.1	23.1±2.8	2.3±1.2	7.2±4.0	80.7±0.8
FAIRSPLIT	**18.2±1.6**	**9.7±1.0**	**22.8±2.8**	**1.8±1.3**	**7.1±3.5**	80.7±0.7

G.7 Fairness-Utility Trade-off and the Base-Rate Gap

To isolate the effect of the base-rate gap, we fix the task (`Smiling` on CelebA, gender as s), the frozen CLIP ViT-L/14 features with a trainable projection, and all hyperparameters including ρ_f. We then resample the four (y, s) cells so that only $\Delta\pi$ changes, holding $\mathbb{P}(y{=}1)$, $\mathbb{P}(s{=}1)$, and the sample size fixed. Table 19 shows that FAIRSPLIT reduces ΔDP by about half at every level, and by more at the smallest gap, while its AP cost grows with $\Delta\pi$.

G.8 Hyperparameter Sensitivity

Figure 6 ablates each of the four loss weights $\lambda_f, \lambda_s, \rho_f, \rho_s$ on CelebA-A while holding the others at their default values. Fairness improvements are driven primarily by the independence weight ρ_f. ΔDP, ABCC, and MCDP all decrease sharply as ρ_f grows, at a gradual utility cost in AP. Varying λ_f has little effect on the metrics. The Sensitive Space weights λ_s and ρ_s both shrink d_f. A larger λ_s also

Table 19: Varying only the base-rate gap $\Delta\pi$ (frozen CLIP ViT-L/14, `Smiling`, 5 seeds).

$\Delta\pi$	AP ↑		ΔDP ↓		AP cost ↓
	ERM	FAIRSPLIT	ERM	FAIRSPLIT	
5	97.5±0.0	96.6±0.3	4.0±0.2	0.7±0.5	0.9
15	97.5±0.0	96.4±0.2	14.4±0.5	6.6±0.7	1.1
25	97.6±0.1	95.3±0.3	24.7±0.3	12.6±1.1	2.3
35	97.8±0.1	93.9±0.3	35.4±0.3	17.4±1.6	3.9
45	98.0±0.1	91.3±0.4	45.7±0.3	20.3±1.9	6.7

lowers ΔDP and ABCC, at a cost in AP, while a larger ρ_s allocates more dimensions to $\mathbf{z}_s$ with almost no change in AP, which tightens the bound in Theorem 3.2. The outcome-conditioned metrics ΔEOpp and ΔEOdd fall as ρ_f grows to its default value and rise slightly beyond it, and they rise with λ_s.

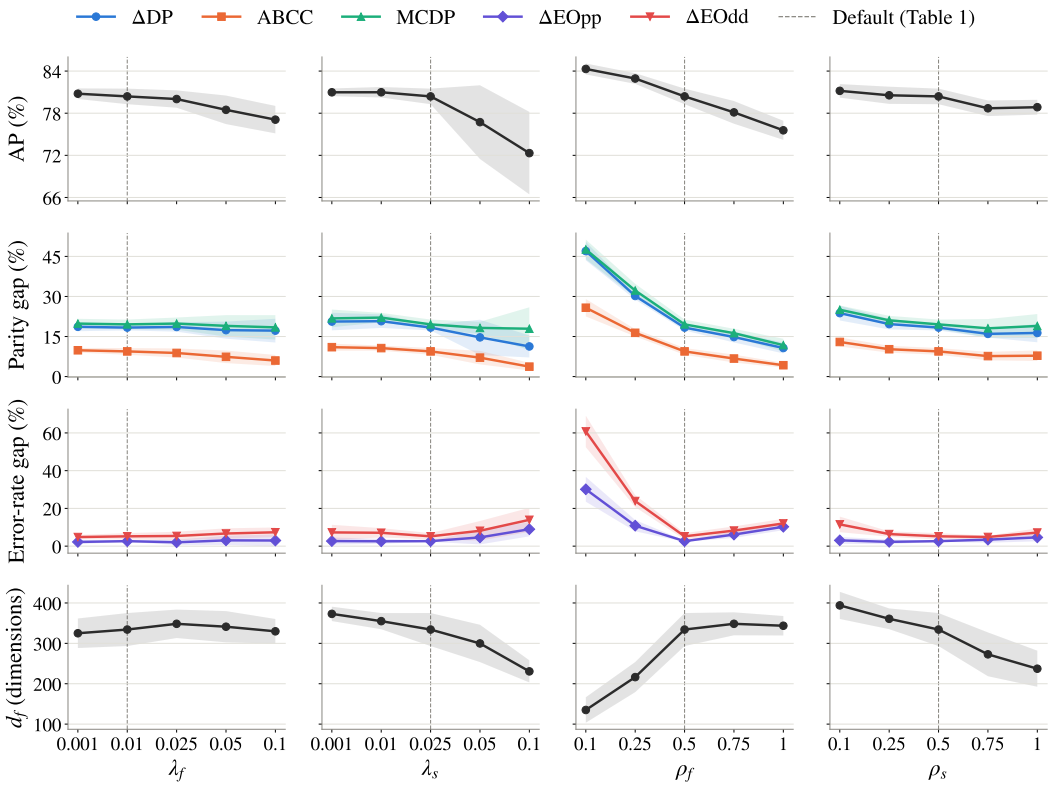


Figure 6: Hyperparameter ablation on CelebA-A (5 seeds). We vary $\lambda_f, \lambda_s, \rho_f, \rho_s$ (left to right) while fixing the other three at their defaults (dashed line). Rows: AP, parity gaps (ΔDP, ABCC, MCDP), error-rate gaps (ΔEOpp, ΔEOdd), and the learned Fair Space dimensionality d_f. Fairness improvements are driven primarily by ρ_f, and the method is robust to λ_f. Increasing λ_s and ρ_s shrinks d_f.

G.9 Component Ablation Across Datasets

Table 20 extends Table 5 to UTKFace-AR and reports utility and all five fairness metrics on both image datasets. The same progression observed on CelebA-A holds on UTKFace-AR. Introducing the mask without any Sensitive Space loss (Unregularized) already reduces ΔDP from 26.6 to 22.2 on CelebA-A and from 17.8 to 15.3 on UTKFace-AR while shrinking d_f, and regularizing the Sensitive Space recovers full FAIRSPLIT's gains. Adding only λ_s or only ρ_s to Unregularized isolates each Sensitive Space term, and FAIRSPLIT without λ_f removes the Fair Space compression. The two Sensitive Space terms contribute comparably, and neither alone reaches the ΔDP of full FAIRSPLIT. Removing λ_f matches full FAIRSPLIT on CelebA-A but costs 1.7 F1 on UTKFace-AR. Removing

ρ_f eliminates the fairness gains on both datasets, and on CelebA-A the Fair Space collapses to 25 dimensions on average.

Table 20: Component ablation on CelebA-A and UTKFace-AR (5 seeds). The utility metric is AP on CelebA-A and F1 on UTKFace-AR. The best value per metric within each dataset is in **bold**. †One of the five seeds collapses to a constant predictor.

Variant	AP/F1↑	ΔDP↓	ABCC↓	MCDP↓	ΔEOpp↓	ΔEOdd↓	d_f
CelebA-A							
ERM	**84.5±0.4**	52.8±0.8	35.9±0.8	53.4±0.8	36.8±1.5	75.0±2.1	512
No Sensitive Space	80.7±0.7	26.6±1.4	16.1±1.0	26.8±1.0	5.5±0.9	18.4±2.8	512
Unregularized (λ_s=ρ_s=0)	80.1±0.8	22.2±2.4	12.4±1.7	22.8±1.6	1.9±0.9	9.1±2.4	391
Unregularized + λ_s	79.9±1.2	20.1±2.7	10.7±1.8	21.6±2.0	2.4±1.3	8.1±2.5	366
Unregularized + ρ_s	80.9±0.6	21.4±1.8	11.2±1.0	22.4±1.2	**1.4±2.2**	6.8±2.8	371
FAIRSPLIT without λ_f	80.6±0.6	18.4±0.9	9.6±0.8	**19.4±1.2**	2.1±0.8	**4.4±1.2**	326
FAIRSPLIT without ρ_f †	82.9±1.2	51.7±26.6	12.1±8.9	75.0±8.8	53.2±28.1	87.4±45.8	25
FAIRSPLIT	80.4±1.1	**18.3±1.1**	**9.4±1.3**	19.6±1.8	2.6±0.3	5.2±1.7	334
UTKFace-AR							
ERM	**73.8±0.7**	27.9±1.4	27.1±2.6	36.2±2.4	29.1±5.2	35.9±3.0	512
No Sensitive Space	71.8±1.6	17.8±2.0	15.7±1.9	23.3±1.9	18.5±4.5	24.3±5.8	512
Unregularized (λ_s=ρ_s=0)	71.1±1.1	15.3±1.1	10.9±0.9	19.9±1.3	19.3±6.1	23.9±6.0	389
Unregularized + λ_s	70.0±1.8	13.3±1.2	**9.0±0.9**	**18.7±1.1**	17.8±6.7	22.7±5.9	223
Unregularized + ρ_s	71.1±1.4	14.0±2.1	11.2±1.2	20.0±1.7	15.9±4.7	**22.1±5.0**	402
FAIRSPLIT without λ_f	68.7±1.7	13.0±2.2	10.3±1.0	19.4±2.2	16.5±6.9	22.2±6.9	222
FAIRSPLIT without ρ_f	**73.8±1.5**	33.4±4.1	19.5±1.1	39.4±2.7	37.8±4.9	49.5±5.1	270
FAIRSPLIT	70.4±1.5	**12.6±1.1**	9.1±1.0	**18.7±1.2**	**14.9±4.6**	**22.1±4.9**	233

G.10 HSIC Kernel and Bandwidth Sensitivity

The HSIC regularizer $\mathcal{L}_{\text{reg}}$ can be instantiated with any characteristic kernel. We test whether FAIRSPLIT is empirically stable across kernel choices. All experiments are on CelebA-A with otherwise identical hyperparameters, averaged over 5 seeds.

Kernel choice. Table 21 compares three characteristic kernel families (Gaussian RBF, Matérn $3/2$, Laplacian). Gaussian and Matérn yield nearly identical results, suggesting stability across smooth characteristic kernels. The Laplacian kernel gives weaker fairness at the same regularization strength, but still reduces ΔDP by 33% relative to ERM.

Bandwidth sensitivity. For the Gaussian RBF kernel, we sweep the bandwidth $\sigma \in \{2, 3, 4, 5, 10\}$ on CelebA-A (Table 21). Every bandwidth reduces ΔDP by 53% to 65% relative to ERM. We fix $\sigma = 5$ for image and $\sigma = 2$ for tabular experiments (Table 11).

G.11 Computational Cost

We measure wall-clock training time per run on a single NVIDIA A100, averaged over 5 seeds, for both a tabular (Adult) and an image (CelebA-A) dataset. Table 22 reports absolute seconds and the relative cost over the ERM baseline. FAIRSPLIT adds 17 to 22% wall-clock overhead over ERM, which lies within the range of the other in-processing fairness methods we evaluate (0.93 to $1.76\times$). On Adult, the overhead comes primarily from the HSIC computation in Equation 4. On CelebA-A, it comes mostly from initializing the mask, which requires one pass over the training set, while the per-step HSIC and mask updates (Proposition 3.1) add only about 3%.

G.12 Robustness to Noisy and Missing Sensitive Attributes

FAIRSPLIT requires sensitive attributes during training, but in practice, these labels can be partially incorrect or unavailable. We assess robustness on CelebA-A under two corruption regimes, without changing any hyperparameters. *Noisy p:* each sensitive label is independently flipped with probability p. *Missing p:* a fraction p of training samples have their sensitive label dropped entirely. For these

Table 21: HSIC kernel family and Gaussian RBF bandwidth on CelebA-A (5 seeds). Only the kernel or σ changes, and ERM is reported as a reference without fairness regularization.

	AP↑	ΔDP↓	ABCC↓
ERM	84.5±0.4	52.8±0.8	35.9±0.8
Kernel family ($\sigma = 5$)			
Gaussian RBF	80.4±1.1	18.3±1.1	9.4±1.3
Matérn $3/2$	80.5±1.5	18.6±2.9	10.0±2.1
Laplacian	78.1±2.5	35.4±6.6	16.4±4.2
Gaussian RBF bandwidth			
$\sigma = 2$	80.8±1.5	24.7±2.4	13.5±1.5
$\sigma = 3$	80.7±0.9	19.9±1.9	10.6±1.6
$\sigma = 4$	81.0±1.0	19.9±1.0	10.4±0.6
$\sigma = 5$	80.4±1.1	18.3±1.1	9.4±1.3
$\sigma = 10$	79.3±1.4	20.8±1.6	9.8±1.5

Table 22: Wall-clock training time per run (in seconds), averaged over 5 seeds on a single NVIDIA A100. "vs. ERM" is the multiplicative cost relative to the ERM baseline.

Method	Adult (s)	vs. ERM	CelebA-A (s)	vs. ERM
ERM	11.4	1.00×	27.7	1.00×
AdvDebias	10.6	0.93×	28.6	1.03×
FairMixup	12.4	1.09×	35.8	1.29×
DRAlign	13.8	1.21×	48.6	1.76×
HSIC(Y, S)	11.9	1.05×	28.4	1.03×
DiffMCDP	16.3	1.43×	28.8	1.04×
PRemover	11.3	0.99×	29.8	1.08×
MMD-B-Fair	16.6	1.46×	35.5	1.28×
FAIRSPLIT	13.9	1.22×	32.3	1.17×

samples, the HSIC and structural-compression terms involving s are masked out, while the cross-entropy task loss is still applied. As a control, we also train with randomly shuffled sensitive labels. Results are averaged over 5 seeds. As shown in Table 23, FAIRSPLIT tolerates missing sensitive labels well: with 10% and 30% of the labels removed, ΔDP is 21.8 and 28.0 against 52.8 for ERM, and the gain shrinks gradually as more labels are removed. It is more sensitive to incorrect labels: 10% label noise keeps most of the gain (ΔDP 28.3), while 30% removes it. With shuffled labels, FAIRSPLIT matches ERM, so its gains come from the sensitive labels. Extending FAIRSPLIT to the semi-supervised setting is left for future work.

Table 23: Robustness on CelebA-A under noisy and missing sensitive attributes (5 seeds). "Noisy p" flips each sensitive label with probability p. "Missing p" drops the sensitive label for a fraction p of samples (these samples contribute only to the cross-entropy loss). "Shuffled" randomly permutes all sensitive labels. ERM and clean FAIRSPLIT are reported as references.

Condition	AP↑	ΔDP↓	ABCC↓
ERM	84.5±0.4	52.8±0.8	35.9±0.8
FAIRSPLIT (clean)	80.4±1.1	18.3±1.1	9.4±1.3
Noisy 10%	82.8±0.7	28.3±2.6	16.2±1.5
Noisy 30%	85.5±0.2	53.1±0.9	31.5±0.9
Noisy 50%	85.6±0.3	54.0±0.4	32.2±0.6
Missing 10%	81.0±0.9	21.8±1.9	11.6±1.2
Missing 30%	81.4±1.2	28.0±1.3	15.4±1.0
Missing 50%	83.3±0.7	36.7±1.7	22.5±0.9
Missing 70%	84.4±0.4	46.5±1.1	29.4±0.4
Shuffled	85.4±0.2	53.4±0.9	31.8±0.6

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our theoretical claim is supported with a proof: we bound a continuous relaxation of the demographic parity gap by the HSIC between the Fair Space and the sensitive attribute and the Fair Space dimensionality, both of which FAIRSPLIT controls during training. Empirically, we validate the method on standard fair classification benchmarks, including multiclass and non-binary sensitive-attributes.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We provide a dedicated *Limitations and Discussion* subsection (Section 3.5) covering the linear-head assumption of our bound, the bound's $1/p_{\min}$ dependence on minimum group prevalence, the requirement for sensitive attributes during training, and the residual sensitive information retained in the Sensitive Space. The first three are paired with targeted experiments and pointers to the relevant appendix. The last is an inherent design property and is addressed by a deployment recommendation.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper has two main theoretical results: the closed-form mask (Proposition 3.1) and the demographic parity bound (Theorem 3.2). Both state their assumptions explicitly in the theorem statements and have complete proofs in Appendix C and Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We describe the experimental setup in Section 4. Section F covers the datasets, metrics, implementation, software and hardware setup, and the used hyperparameters for every method. Section F.1 also provides per-dataset licenses and download sources. Our code, including all training configurations, is publicly available at `https://github.com/shkabrii/fairsplit`.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Our code is publicly available at `https://github.com/shkabrii/fairsplit`, with instructions to reproduce the main experiments for FAIRSPLIT and all baselines. Section F.1 provides per-dataset licenses and download sources, and Section F documents the environment, the used hyperparameters, and the experimental protocol.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4 specifies datasets, baselines, evaluation metrics, the five-seed protocol, and the model-selection rule. Section F gives the optimizer, learning-rate schedule, encoder architectures, batch sizes, training length, and the per-method hyperparameters.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All main tables report mean $\pm$ one standard deviation across five seeds (Section 4), with seed variation covering the train/val/test split, model initialization, and batch order. We do not perform dedicated significance tests.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section F.4 reports the hardware. Section G.11 provides per-run wall-clock training times for FAIRSPLIT and all baselines on Adult and CelebA-A, averaged over 5 seeds.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We read and followed the NeurIPS Code of Ethics. The work targets fairness directly (mitigating disparate treatment based on sensitive attributes), uses only established public benchmarks under their stated terms (Section F.1) and involves no human-subjects experimentation.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The work targets a positive societal application (mitigating disparate treatment based on sensitive attributes), discussed in Section 1 and Section 2. Specific risks and limitations are addressed in Section 3.5: the Sensitive Space retains protected-attribute information so model parameters must be secured, and the method requires sensitive labels at training time, with partial robustness to noisy or missing labels shown in Section G.12.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not release pre-trained models or data.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets are cited and their licenses, source URLs, and terms of use are documented in Section F.1. Code we build upon (FFB [22], DiffMCDP [28], H-SPLID [41], MMD-B-Fair [11]) is cited with repository links in Section F, and we use these assets in accordance with their stated terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We release our code at `https://github.com/shkabrii/fairsplit` with a README documenting setup, dependencies, and commands to reproduce the main experiments.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No research with human subjects or crowdsourcing.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No research with human subjects or crowdsourcing.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were only used for writing, editing, formatting, and coding assistance.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.