Supplementary Materials

A. Additional Experimental Results

An extensive overview for the results visualized in Figure 4 are listed in Table S1.

B. GAN based SMOTE

Algorithm S2 GAN based synthetic minority oversampling

Inputs:
Hyper-parameters $m, k \in \mathbb{N}^+$
Train dataset $D_{\text{Train}} = \{(x_0, y_0), (x_1, y_1), \dots\}, x_i \in \mathcal{X}, y_i \in \mathcal{Y}$
Trained invertible GAN $E : \mathcal{X} \to \mathcal{Z}, G : \mathcal{Z} \to \mathcal{X}, Z = \mathbb{R}^n$
datapoint (x_s, y_s)
Encode dataset
$D_{\text{Train}} \leftarrow \{(x_0, y_0, z_0), (x_1, y_1, z_1), \dots\}, z_i = E(x_i)$
Determine m nearest neighbours
$N \leftarrow \{z_0^N, z_1^N, \dots, z_m^N\}$
according to $z_i^N = \arg\min_{z_j \notin \{z_0^N, \dots, z_{i-1}^N\}} \|z_s - z_j\|^2$ s.t. $y_s = y_j$
Randomly choose k neighbours from that set
$N \leftarrow \{x_{\pi(0)}^N, x_{\pi(1)}^N, \dots, x_{\pi(k)}^N\}$
Uniformly sample $z_{\text{Augment}} \sim \mathcal{U}\,[\text{Simplex}\,[N]]$
Return $(G(z_{\text{Augment}}), y_s)$

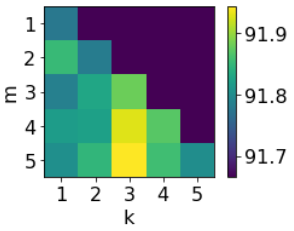

Figure S1. The effect of varying the number of additional neighbors used in the simplex (k) against the number of close neighbors sampled from. We see strongest results where $k = 3$ (note that k is the number of neighbors, so in this case it corresponds to a 4-simplex), and where we sample from a greater range of neighbors. $k = 1$ corresponds to standard SMOTE.

C. Efficient Uniform sampling from the Convex Hull of Points

Here, we briefly summarize the strategy for efficiently uniformly sampling from the convex hull of a set of k points in an n-dimensional space where $k \leq n + 1$, as we have only been able to find related approaches in the literature for $k = 3$ [60].

Under minimal assumptions, e.g., the points are sampled with continuous noise, with probability 1 the k points form a $k - 1$ dimensional simplex, or generalized triangle. This allows us to sample efficiently using an inductive process. Given a point ρ_i sampled from a i dimensional simplex, S_i we can uniformly sample from an $i+1$ dimensional simplex,

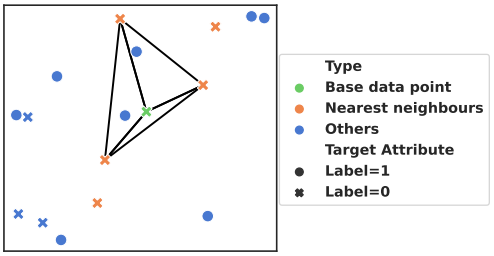

Figure S2. Visualization of the uniform simplex sampling strategy for improved data diversity. Given a datapoint (green) and its nearest m neighbours with the same target label (orange) we randomly choose k neighbours to form a simplex. datapoints with other labels (blue crosses) or too large distance (blue dots) are ignored.

$S_{i+1} = S_i \cup \{p_i\}$ by choosing

$$\rho_{i+1} = \lambda^{1/i}\rho_i + (1 - \lambda^{1/i})p_{i+1} \tag{S1}$$

where λ is sampled from the uniform distribution $U[0, 1]$.

The base case is $S_1 = \{p_1\}$, and $\rho_1 = p_1$.

Note that when the assumption fails, and points are co-linear and do not form a simplex, the algorithm degrades reasonably. Sampled points will lie inside the convex hull of points, but the samples will not be uniform.

D. Architecture, Dataset and Details

The classifiers are built upon pretrained ResNet50 models [39]. For each of the binary attributes of the CelebA dataset, one classifier is trained. As the protected attribute we chose *"male"*[7]. Each model was trained for $3 \cdot 10^6$ images and evaluated every 500 batches using Adam ($lr = 10^{-4}$) and a batch size of 64. Images were center-cropped and down-scaled to 128×128. We use RandAugment with $N = 3$, $M = 15$ for every experiment unless otherwise stated. The reported numbers are the retrieved peak performance during the training period, evaluated on the hold-out evaluation dataset. We rely on the analysis of [66] and report the means over the attributes with *gender independent label quality*: "bags under eyes", "bangs", "black hair", "blond hair", "brown hair", "chubby", "eyeglasses", "gray hair", "high cheekbones", "mouth slightly open", "narrow eyes", "smiling", "wearing hat". Rows labelled with * show results achieved using the codebase of [82]. For the invertible GAN model, we choose InvGAN [32], however the requirements on the model are

[7]The labels in CelebA refer to an externally assigned perceived binary gender, and not to self-assigned gender identity. Although the binary nature of the label does not reflect the true distribution of either, we are restricted to the annotations available in the dataset.

Method	Baseline multi task [82]	Weighting [82]	Domain Disc. [70, 82]	Domain Indep. [82]	Uniconf. Adv. [4]	Baseline single task [66]	GAN Debiasing [66]	Regularized [61,83]	g-SMOTE + Adaptive Sampling [ours]	g-SMOTE [ours]	Baseline FairMixup [20]	FairMixup [20]
Accuracy	**91.79**	91.45	91.78	91.24	90.86	92.47	92.12	91.05	92.56	**92.64**	**92.74**	88.46
Max. grp. acc.	93.66	93.35	**93.69**	93.04	93.15	94.46	94.03	94.42	94.44	**94.59**	**93.85**	90.42
Min. grp. acc.	**89.52**	89.06	89.39	88.93	88.08	90.14	89.85	87.86	**90.36**	90.35	**91.44**	86.36
TPR	64.51	64.02	62.80	**70.74**	50.15	**67.90**	66.13	54.20	67.11	66.14	**79.13**	46.67
Max. grp. TPR	72.29	67.41	70.13	**75.61**	59.59	73.88	70.36	56.11	**74.06**	73.43	**80.89**	47.85
Min. grp. TPR	57.09	59.74	55.06	**66.05**	40.72	**61.34**	61.25	52.34	59.78	58.32	**72.92**	44.27
DEO	15.20	**7.67**	15.08	9.56	18.87	12.54	9.11	**3.77**	14.28	15.11	7.97	**3.58**
DEODD	18.14	**9.00**	18.10	13.29	21.48	16.54	12.04	**5.06**	19.30	19.32	10.06	**4.29**

Table S1. Fairness methods on the CelebA dataset. We report mean scores over the 13 labels [66] call *gender independent*. We report the model with the greatest min. group accuracy over the training period.

Attribute Name	Baseline	AdaptiveSMOTE
Wavy Hair	76.95 ± 0.32	77.86 ± 0.11
Big Lips	80.49 ± 0.15	80.30 ± 0.16
Eyeglasses	99.23 ± 0.01	99.25 ± 0.02
Attractive	78.80 ± 0.14	79.17 ± 0.10
Brown Hair	81.26 ± 0.05	81.18 ± 0.12
Wearing Necklace	80.30 ± 0.01	80.30 ± 0.01
High Cheekbones	85.53 ± 0.11	85.61 ± 0.22
Receding Hairline	91.08 ± 0.09	91.49 ± 0.11
Wearing Hat	98.21 ± 0.02	98.22 ± 0.08
Black Hair	86.40 ± 0.10	87.19 ± 0.26
Gray Hair	95.28 ± 0.09	95.39 ± 0.08
Pale Skin	95.26 ± 0.06	95.43 ± 0.07
Smiling	91.64 ± 0.17	91.91 ± 0.09
Chubby	89.29 ± 0.02	89.35 ± 0.12
Young	82.28 ± 0.10	82.99 ± 0.24
Wearing Earrings	83.46 ± 0.14	83.67 ± 0.23
Big Nose	70.71 ± 0.29	71.12 ± 0.32
Oval Face	70.67 ± 0.10	71.15 ± 0.03
Bags Under Eyes	73.24 ± 0.37	73.69 ± 0.19
Bushy Eyebrows	87.08 ± 0.09	87.36 ± 0.08
Mouth Slightly Open	93.46 ± 0.12	93.58 ± 0.06
Rosy Cheeks	90.87 ± 0.11	90.97 ± 0.09
Arched Eyebrows	76.67 ± 0.35	76.83 ± 0.26
Blurry	95.58 ± 0.05	95.61 ± 0.06
Wearing Lipstick	86.14 ± 0.17	86.44 ± 0.14
Blond Hair	91.96 ± 0.05	92.10 ± 0.13
Heavy Makeup	84.13 ± 0.36	84.13 ± 0.01
Pointy Nose	69.20 ± 0.16	69.86 ± 0.11
Straight Hair	77.72 ± 0.42	77.98 ± 0.12
Bangs	94.67 ± 0.17	94.72 ± 0.04
Double Chin	91.43 ± 0.04	91.57 ± 0.17
Narrow Eyes	91.97 ± 0.08	92.32 ± 0.21

Table S2. Min. group accuracy for individual attributes of CelebA. Reported are the means and standard deviations over 3 restarts of all attributes with at least 11 positive and negative datapoints per group. The protected attribute is *"male"*. We evaluate the model with greatest min. group accuracy over the training period. Both methods were trained on 10000 images from the CelebA training set.

low: Aside from state-of-the-art image quality, we need decent interpolation capabilities as well as a reasonable semantic structure of the latent space.

Regularized Approach. The regularized model presented in Figure 4 is based on a regularizer used in [83]. Given training data $\mathcal{D} = \{x_i, y_i, a_i\}_{i=1}^n$ and a model f (as above) that outputs classification scores in $[0, 1]$, we define the regularizer $\mathcal{R}^{\text{DEO}}$ as

$$\mathcal{R}^{\text{DEO}}(f) := \left(\frac{1}{n_{11}} \sum_{\mathcal{D}_{11}} f(x_i) - \frac{1}{n_{10}} \sum_{\mathcal{D}_{10}} f(x_i) \right)^2,$$

where $\mathcal{D}_{ya} = \{(\tilde{x}, \tilde{y}, \tilde{a}) : \tilde{a} = a, \tilde{y} = y, (\tilde{x}, \tilde{y}, \tilde{a}) \in \mathcal{D}\}$ and $n_{ya} = |\mathcal{D}_{ya}|$ for $g \in \{0, 1\}$.

We simply add the fairness regularizer to the loss and trade-off fairness with the accuracy of the classifier via a hyperparameter λ. We minimize $\widehat{\mathrm{L}}(f) = \sum_{i=1}^n l(f(x_i), y_i) + \lambda \mathcal{R}^{\text{DEO}}(f)$.