Beyond Pattern Recognition
Face Recognition Systems - Similarity Scoring

13 juin 2023

So far, we have seen :

- how to formulate the **Pattern Recognition** problem by means of notions of **Probability/Statistics**,
- many **use cases** (e.g. medical diagnosis tools, computer vision, audio analysis),
- many popular algorithms (e.g. ensemble learning, deep learning, SVM) and that notions of **Optimization** are required to undestand them
- that these problems are more **difficult to solve in practice** than to state (cf model selection/assessment)

Pattern Recognition is the **flagship problem** in AI (ML) :

- it is well understood from a **theoretical perspective** (cf Vapnik-Chervonenkis theory)
- the task is **ubiquitous**
- many **reliable softwares** can be used
- **many other tasks** are performed by machines using variants of it (e.g. biometrics, recommending systems).

Today (and during next lab), this will be illustrated by **Face Recognition**

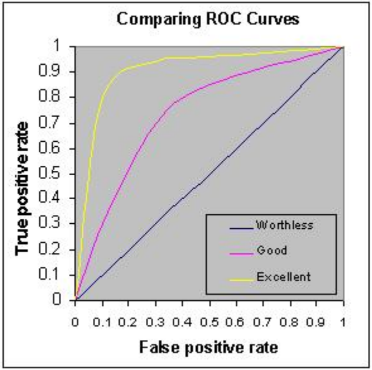
- **Motivation :** many applications of Face Recognition, e.g. access control, identity verification (smartphones), social media ...
- **Threats :** bias with respect to race, gender, age
- **Different causes of bias**, a very hot topic in AI now

Table of Contents

Preliminaries on ROC Analysis

- Same data as in binary classification
- Posterior probability $\eta(x) = \{Y = 1 \mid X = x\}$
- Ordering on $\mathbb{R}^d$ defined through scoring $s : \mathbb{R}^d \to \mathbb{R}$
- Goal : build an program $s(x)$ so as rank the elements of $\mathbb{R}^d$ as η : $x \preceq x'$ iff $s(x) \leq s(x')$

Scoring and ROC curves



- True Positive rate :

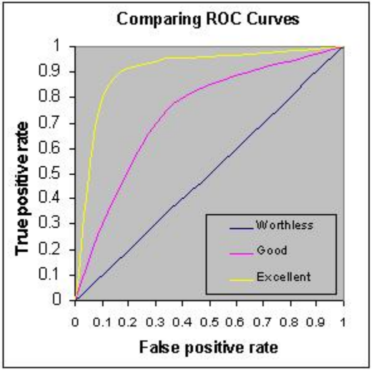
$$\mathrm{tpr}_s(t) = (s(X) \geq t \mid Y = 1)$$

- False Positive Rate :

$$\mathrm{fpr}_s(t) = (s(X) \geq t \mid Y = -1)$$

Receiving Operator Characteristic : $\quad t \mapsto \left(\mathrm{fpr}_s(t), \mathrm{tpr}_s(t)\right)$

Scoring and ROC curves



- True Positive rate :

$$\text{tpr}_s(t) = (s(X) \geq t \mid Y = 1)$$

- False Positive Rate :

$$\text{fpr}_s(t) = (s(X) \geq t \mid Y = -1)$$

Receiving Operator Characteristic : $\quad t \mapsto \left(\text{fpr}_s(t), \text{tpr}_s(t)\right)$

AUC = Area Under the ROC Curve

The data can be modeled by i.i.d. realizations of a random variable $(X, Y) \in \mathbb{R}^{h\times w\times c} \times \{1, \ldots, K\}$ with law $\mathbb{P}$. Dataset : $(\boldsymbol{x}_i, y_i)_{1\leq i\leq N}$.

Goal : learn an encoder function $f_\theta : \mathbb{R}^{h\times w\times c} \to \mathbb{R}^d$ that embeds the images in a way to bring same identities closer together.

$Z := f_\theta(X)$ is the *face embedding* of X.

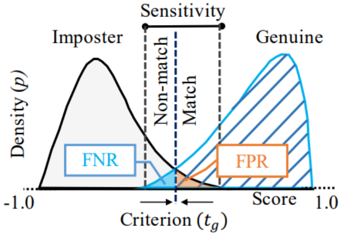
The closeness between two embeddings is usually quantified with the cosine similarity measure $s(\boldsymbol{z}_i, \boldsymbol{z}_j) := \boldsymbol{z}_i^\intercal \boldsymbol{z}_j / (||\boldsymbol{z}_i|| \cdot ||\boldsymbol{z}_j||)$.

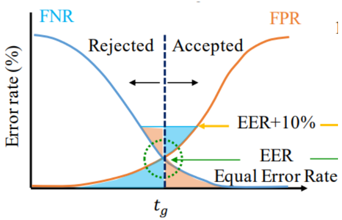
An operating point $t \in [-1, 1]$ (threshold of acceptance) has to be chosen to classify a pair $(\boldsymbol{z}_i, \boldsymbol{z}_j)$ as :

- *genuine* (same identity) if $s(\boldsymbol{z}_i, \boldsymbol{z}_j) \geq t$
- *impostor* (distinct identities) if $s(\boldsymbol{z}_i, \boldsymbol{z}_j) < t$.

Evaluation metrics. Let (X_1, y_1) and (X_2, y_2) be two i.i.d. random variables with law $\mathbb{P}$. We distinguish between the False Acceptance and False Rejection Rates, respectively defined by :

$$\mathrm{FAR}(t) := \mathbb{P}(s(Z_1, Z_2) \geq t \mid y_1 \neq y_2)$$
$$\mathrm{FRR}(t) := \mathbb{P}(s(Z_1, Z_2) < t \mid y_1 = y_2)$$

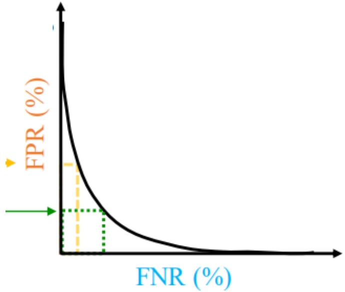




Canonical metric : For $\alpha \in [0, 1]$,

$$\mathrm{FRR@(FAR} = \alpha) := \mathrm{FRR}(t) \qquad \text{with } t \text{ s.t. } \mathrm{FAR}(t) = \alpha.$$

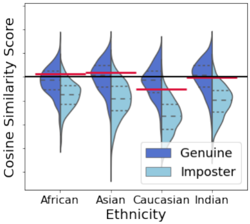
$\rightarrow$ ROC/DET curve.

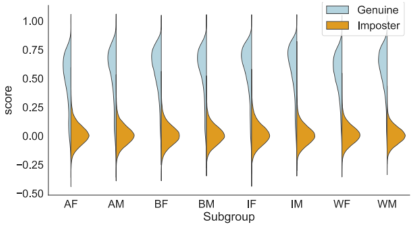


$\rightarrow$ In biometrics, we are interested in $\mathrm{FRR@(FAR} = \alpha)$ for $\alpha = 10^{-i}$ with $i \in \{1, ..., 6\}$.

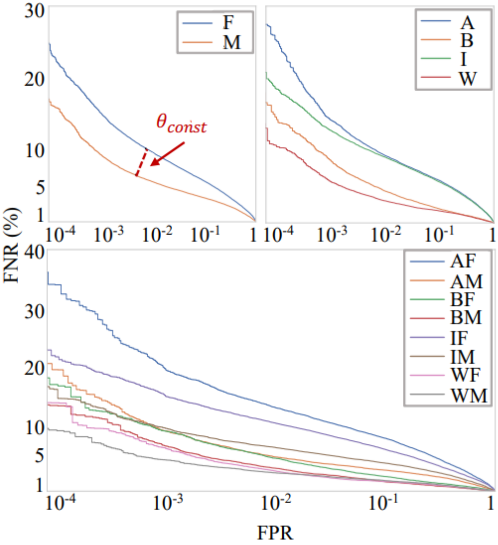
Fairness. Discrete sensitive attribute that can take $A > 1$ different values.
$\rightarrow$ random variables (X_i, y_i, a_i) where $a_i \in \mathcal{A} = \{0, 1, \ldots, A-1\}$.
For $a \in \mathcal{A}$:

$$\mathrm{FAR}_a(t) := \mathbb{P}(s(Z_1, Z_2) \geq t \mid y_1 \neq y_2,\ a_1 = a_2 = a)$$
$$\mathrm{FRR}_a(t) := \mathbb{P}(s(Z_1, Z_2) < t \mid y_1 = y_2,\ a_1 = a_2 = a).$$

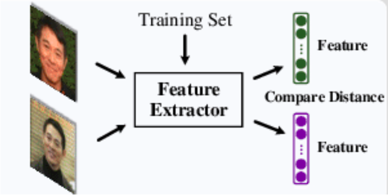




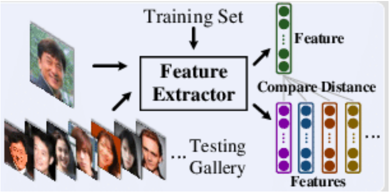
Intra-group ROC curves.



- Face verification :



- Face identification :



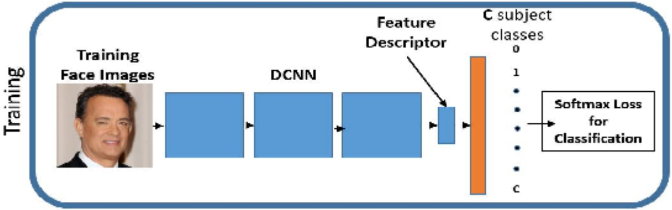


Figure – Workflow of Deep Face Recognition training (**crystal_loss**).

$$i\text{-th image} \xrightarrow{DCNN} \boldsymbol{f_i} \in \mathbb{R}^d \xrightarrow{MLP} \boldsymbol{W}\boldsymbol{f_i} + \boldsymbol{b} = \begin{bmatrix} \boldsymbol{w_1}^\top \boldsymbol{f_i} + b_1 \\ \boldsymbol{w_2}^\top \boldsymbol{f_i} + b_2 \\ \vdots \\ \boldsymbol{w_C}^\top \boldsymbol{f_i} + b_C \end{bmatrix} \xrightarrow{softmax} \boldsymbol{p_i} \in \mathbb{R}^C$$

$$
i\text{-th image} \xrightarrow{DCNN} \boldsymbol{f_i} \in \mathbb{R}^d \xrightarrow{MLP} \boldsymbol{W}\boldsymbol{f_i} + \boldsymbol{b} = \begin{bmatrix} \boldsymbol{w_1}^\top \boldsymbol{f_i} + b_1 \\ \boldsymbol{w_2}^\top \boldsymbol{f_i} + b_2 \\ \vdots \\ \boldsymbol{w_C}^\top \boldsymbol{f_i} + b_C \end{bmatrix} \xrightarrow{softmax} \boldsymbol{p_i} \in \mathbb{R}^C
$$

$$
\mathcal{L}_{\text{softmax}} = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{\boldsymbol{w_{y_i}}^\top \boldsymbol{f_i} + b_{y_i}}}{\sum_{k=1}^{C} e^{\boldsymbol{w_k}^\top \boldsymbol{f_i} + b_k}}
$$

Loss functions

- $\mathcal{L}_{\text{softmax}} = -\frac{1}{N} \sum_{i=1}^{N} \log \dfrac{e^{\boldsymbol{w_{y_i}}^\top \boldsymbol{f_i} + b_{y_i}}}{\sum\limits_{k=1}^{C} e^{\boldsymbol{w_k}^\top \boldsymbol{f_i} + b_k}}$

- $\mathcal{L}_{\text{modified}} = -\frac{1}{N} \sum_{i=1}^{N} \log \dfrac{e^{\kappa \frac{\boldsymbol{w_{y_i}}^\top}{\|\boldsymbol{w_{y_i}}\|_2} \frac{\boldsymbol{f_i}}{\|\boldsymbol{f_i}\|_2}}}{\sum\limits_{k=1}^{C} e^{\kappa \frac{\boldsymbol{w_k}^\top}{\|\boldsymbol{w_k}\|_2} \frac{\boldsymbol{f_i}}{\|\boldsymbol{f_i}\|_2}}}$ **(normface)**

$$= -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{\kappa \, \boldsymbol{\mu_{y_i}}^\top \boldsymbol{x_i}}}{\sum\limits_{k=1}^{C} e^{\kappa \, \boldsymbol{\mu_k}^\top \boldsymbol{x_i}}} \qquad \|\boldsymbol{x_i}\|_2 = \|\boldsymbol{\mu_k}\|_2 = 1$$

- Large-margin loss functions

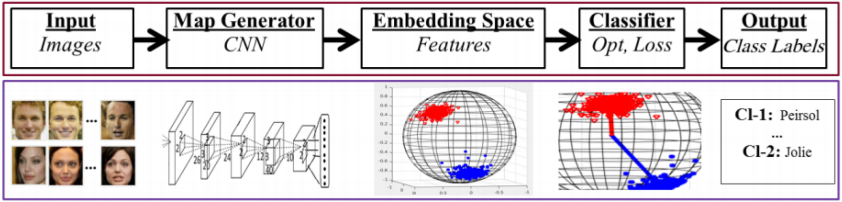


Figure – Workflow of a traditional classifier of faces (**vmf**).

$$\mathcal{L}_{\text{modified}} = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{\kappa \, \boldsymbol{\mu}_{\boldsymbol{y_i}}{}^{\top} \boldsymbol{x_i}}}{\sum_{k=1}^{C} e^{\kappa \, \boldsymbol{\mu}_{\boldsymbol{k}}{}^{\top} \boldsymbol{x_i}}} \qquad \|\boldsymbol{x_i}\|_2 = \|\boldsymbol{\mu_k}\|_2 = 1$$

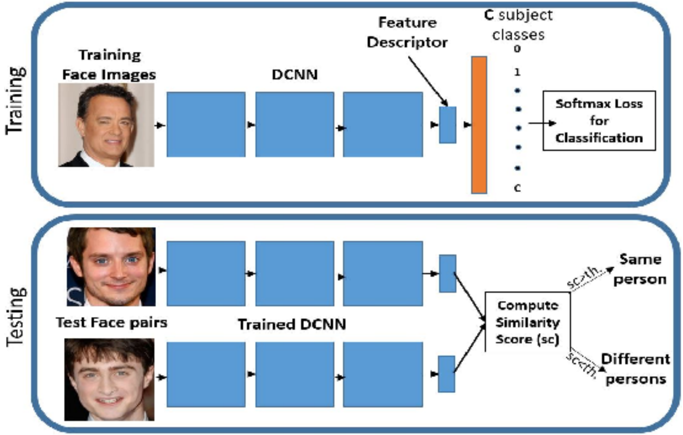


Figure – Workflow of Deep Face Recognition (**crystal_loss**).

$$\text{Similarity score}:\ s(\boldsymbol{f_i}, \boldsymbol{f_j}) = \frac{\boldsymbol{f_i}^\top \boldsymbol{f_j}}{\|\boldsymbol{f_i}\|_2 \ \|\boldsymbol{f_j}\|_2} = \boldsymbol{x_i}^\top \boldsymbol{x_j} = \cos(\theta_{i,j})$$

Evaluation step

- Testing set of n face images $\rightarrow$ score matrix $\mathcal{S} = \big(s(\boldsymbol{f_i}, \boldsymbol{f_j})\big)_{1\leq i,j\leq n}$

- 2 types of scores : $\mathcal{G} = \{\mathcal{S}_{i,j} \mid y_i = y_j,\ i \neq j\}$ (genuines)

  $\mathcal{I} = \{\mathcal{S}_{i,j} \mid y_i \neq y_j\}$ (impostors)

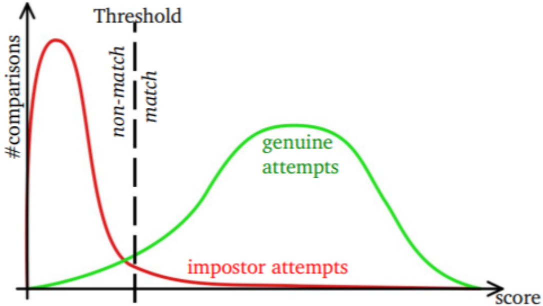


Figure – Typical matching score distribution (**matching**).

Evaluation metrics

- False Acceptance Rate : $\mathrm{FAR}(t) =$ proportion of impostors matched
- False Rejection Rate : $\mathrm{FRR}(t) =$ proportion of genuines non-matched

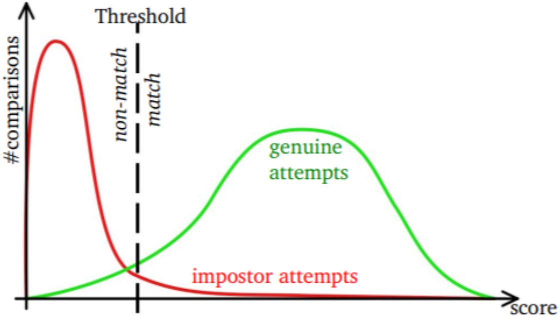


Figure – Typical matching score distribution (**matching**).

Evaluation metrics

- False Acceptance Rate : $\text{FAR}(t) =$ proportion of impostors matched
- False Rejection Rate : $\text{FRR}(t) =$ proportion of genuines non-matched

- FRR@FAR=α : $\text{FRR}(t_\alpha)$ with t_α s.t. $\text{FAR}(t_\alpha) = \alpha$

- DET curve : FRR@FAR=α for $\alpha \in [0, 1]$

DET/ROC curve

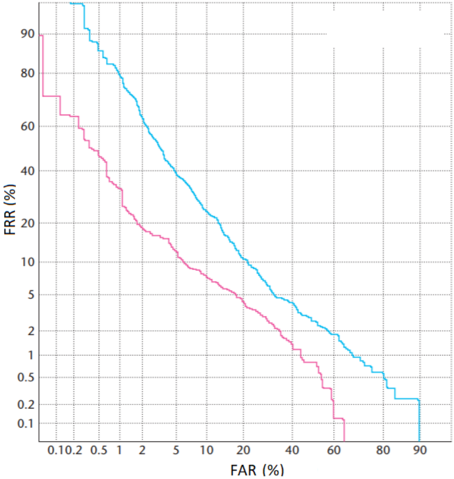


Figure – Two typical DET curves (**det_example**).

Bias in Face Recognition

- Testing phase of Face Recognition is a binary classification with 2 input images ($\hat{Y} = 0 \rightarrow$ no match, $\hat{Y} = 1 \rightarrow$ match)

- Demographic parity :
  $\mathbb{P}(\hat{Y}|A = a) = \mathbb{P}(\hat{Y}|A = a')$
- Equal Opportunity :
  $\mathbb{P}(\hat{Y} = 1|Y = 1, A = a) = \mathbb{P}(\hat{Y} = 1|Y = 1, A = a')$
  $\rightarrow$ Equality of TPR (or equivalently FNR)
- Equalized Odds :
  $\mathbb{P}(\hat{Y} = 1|Y = y, A = a) = \mathbb{P}(\hat{Y} = 1|Y = y, A = a')$
  $\rightarrow$ Equality of TPR (or equivalently FNR) and FPR (or equivalently TNR)

Different accuracy across groups

- Compute metrics of performance for each group g
  $\rightarrow$ Consider the group g in the testing set, get the matching scores
  $\rightarrow \mathrm{FAR}_g(t), \mathrm{FRR}_g(t)$

Different accuracy across groups

- Compute metrics of performance for each group g
  $\rightarrow$ Consider the group g in the testing set, get the matching scores
  $\rightarrow$ $\mathrm{FAR}_g(t), \mathrm{FRR}_g(t)$

- FRR_g@FAR_g=α $\rightarrow$ DET curve for each groupe g

Different accuracy across groups

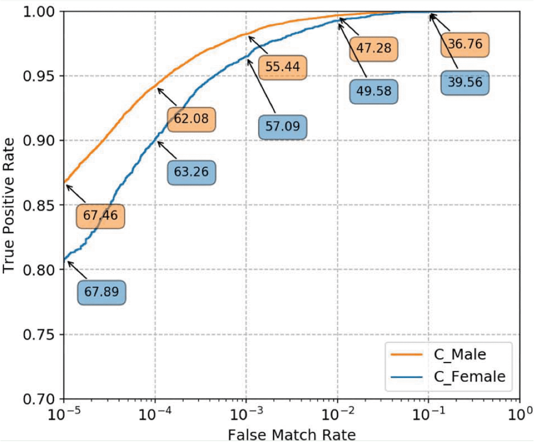


Figure – ROC curves with male/female breakdown (**vggface2**).

Reference group

- Use a group as reference *ref*
  $\rightarrow \text{FAR}_g @\text{FAR}_{ref} = \alpha$
  $\rightarrow \text{FRR}_g @\text{FAR}_{ref} = \alpha$

  $\rightarrow \text{FRR}_g @\text{FRR}_{ref} = \alpha$

Reference group

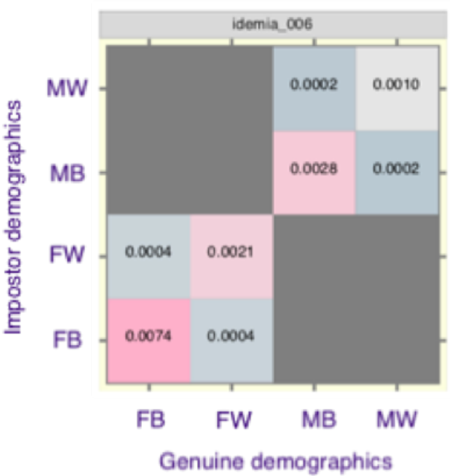


Figure – $FAR_g@FAR_{MW}=10^{-3}$ (**nist_report_2020**).

Global population as reference

- Use a group as reference *ref* :
  $\rightarrow$ $\text{FAR}_g @\text{FAR}_{ref} = \alpha$
  $\rightarrow$ $\text{FRR}_g @\text{FAR}_{ref} = \alpha$
- Use the global population as reference :
  $\rightarrow$ $\text{FAR}_g @\text{FAR} = \alpha$
  $\rightarrow$ $\text{FRR}_g @\text{FAR} = \alpha$

Global population as reference

- Use a group as reference *ref* :
  $\rightarrow$ $\mathrm{FAR}_g @ \mathrm{FAR}_{ref} = \alpha$
  $\rightarrow$ $\mathrm{FRR}_g @ \mathrm{FAR}_{ref} = \alpha$

- Use the global population as reference :
  $\rightarrow$ $\mathrm{FAR}_g @ \mathrm{FAR} = \alpha$
  $\rightarrow$ $\mathrm{FRR}_g @ \mathrm{FAR} = \alpha$

- Metrics of bias :
  $\rightarrow$ $|\Delta \mathrm{FAR}_g @ \mathrm{FAR} = \alpha| \;=\; |\mathrm{FAR}_\mathrm{F} @ \mathrm{FAR} = \alpha - \mathrm{FAR}_\mathrm{M} @ \mathrm{FAR} = \alpha|$
  $\rightarrow$ $|\Delta \mathrm{FRR}_g @ \mathrm{FAR} = \alpha| \;=\; |\mathrm{FRR}_\mathrm{F} @ \mathrm{FAR} = \alpha - \mathrm{FRR}_\mathrm{M} @ \mathrm{FAR} = \alpha|$

Global population as reference

- Use a group as reference *ref* :
  $\rightarrow \mathrm{FAR}_g @\mathrm{FAR}_{ref}{=}\alpha$
  $\rightarrow \mathrm{FRR}_g @\mathrm{FAR}_{ref}{=}\alpha$

- Use the global population as reference :
  $\rightarrow \mathrm{FAR}_g @\mathrm{FAR}{=}\alpha$
  $\rightarrow \mathrm{FRR}_g @\mathrm{FAR}{=}\alpha$

- Metrics of bias :
  $\rightarrow \frac{\mathrm{FAR}_\mathrm{F} @\mathrm{FAR}=\alpha}{\mathrm{FAR}_\mathrm{M} @\mathrm{FAR}=\alpha} := \frac{\mathrm{FAR}_\mathrm{F}}{\mathrm{FAR}_\mathrm{M}} @\mathrm{FAR}{=}\alpha$

  $\rightarrow \frac{\mathrm{FRR}_\mathrm{F} @\mathrm{FAR}=\alpha}{\mathrm{FRR}_\mathrm{M} @\mathrm{FAR}=\alpha} := \frac{\mathrm{FRR}_\mathrm{F}}{\mathrm{FRR}_\mathrm{M}} @\mathrm{FAR}{=}\alpha$

Other bias metrics

- Difference in accuracy
- $|\Delta \mathrm{TAR}_g @\mathrm{FAR}{=}\alpha| \;=\; |\mathrm{TAR}_\mathrm{F} @\mathrm{FAR}{=}\alpha - \mathrm{TAR}_\mathrm{M} @\mathrm{FAR}{=}\alpha|$ **(agenda_gan_gender)**
- $\mathrm{MAD}(\mathrm{TAR}) = \mathbb{E}[\mathrm{TAR}_\mathrm{g} @\mathrm{FAR}{=}\alpha - \mathbb{E}[\mathrm{TAR}_\mathrm{g} @\mathrm{FAR}{=}\alpha]]$ **(comparison_level_race_bias)**
- $\dfrac{\max_g \mathrm{FAR}_g(t)}{\min_g \mathrm{FAR}_g(t)}$
- $\dfrac{\max_g \mathrm{FRR}_g(t)}{\min_g \mathrm{FRR}_g(t)}$ **(nist_prez_demograpics)**

$\rightarrow$ 'Fairness - utility' tradeoff

Framework

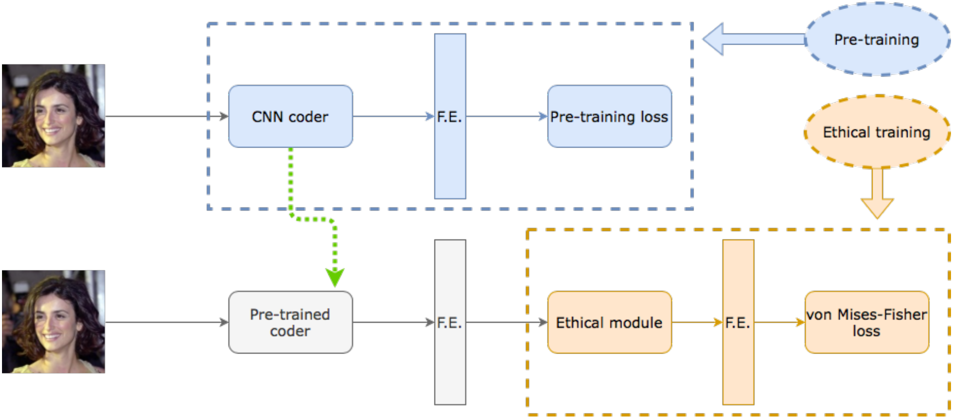


Figure – Proposed approach to reduce gender bias (F.E. = Face Embedding).

Training set of pre-trained model

2M images, 90k labelled identities

Gender statistics :

- Number of images : 60% F, 40% M
- Number of identities : 35% F, 65% M

Genders are determined with a classifier.

Ethical module

Input : feature vectors of dimension 256

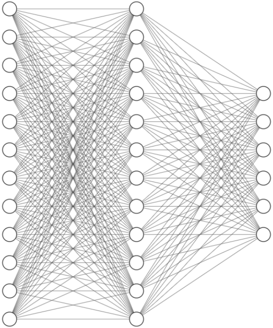


Figure – Toy model MLP : 256 units, 256 units, ReLU activation, d units.

Output : feature vectors $\boldsymbol{f} \in \mathbb{R}^d \rightarrow$ classified with von Mises-Fisher loss

von Mises-Fisher distribution

Density function : $\boldsymbol{x_i} \in \mathbb{R}^d$, $\|\boldsymbol{x_i}\|_2 = 1$

$$V_d(\boldsymbol{x_i}|\boldsymbol{\mu}, \kappa) = C_d(\kappa)\ e^{\kappa\ \boldsymbol{\mu}^\top \boldsymbol{x_i}}$$
$$= C_d(\kappa)\ e^{\kappa\ \cos(\theta)}$$

$\boldsymbol{\mu} \in \mathbb{R}^d$, $\|\boldsymbol{\mu}\|_2 = 1$

$$C_d(\kappa) = \frac{\kappa^{\frac{d}{2}-1}}{(2\pi)^{\frac{d}{2}} I_{\frac{d}{2}-1}(\kappa)}$$

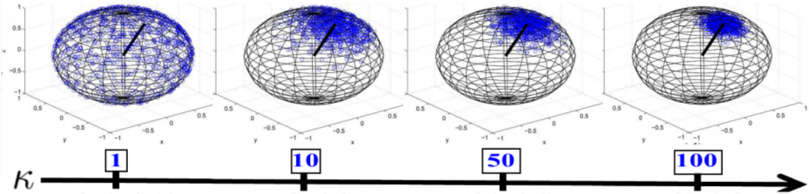
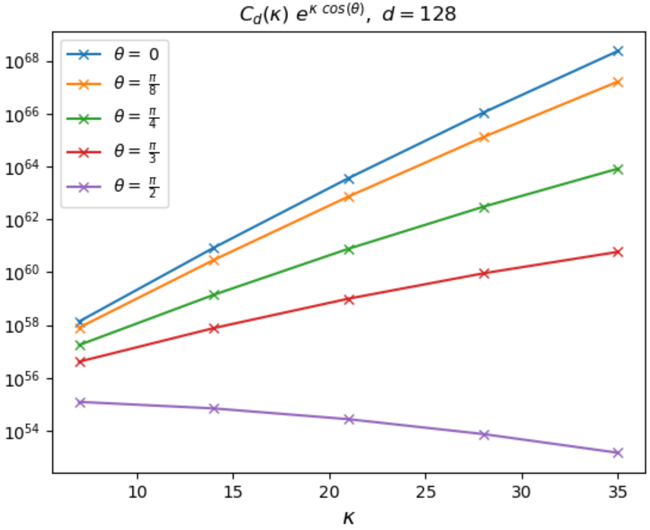


Figure – Samples from vMF distribution, $d = 3$ (**vmf**).

→ Constrained gaussian

Likelihood of von Mises-Fisher distribution



vMF mixture model

Mixture model with C classes : $\boldsymbol{x_i}, \boldsymbol{\mu_j} \in \mathbb{R}^d$, $\|\boldsymbol{x_i}\|_2 = \|\boldsymbol{\mu_j}\|_2 = 1$

$$g_d(\boldsymbol{x_i}|\Theta_C) = \sum_{j=1}^{C} \pi_j \; V_d(\boldsymbol{x_i}|\boldsymbol{\mu_j}, \kappa_j) \qquad \Theta_C = \{(\pi_j, \boldsymbol{\mu_j}, \kappa_j)\}_{1 \le j \le C}$$

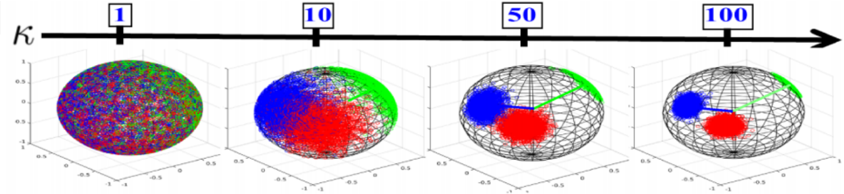


Figure – Samples from vMF mixture model, $d = 3$, $C = 3$ (**vmf**).

Set a statistical model on face embeddings

- Mixture model with C classes : $\boldsymbol{x_i}, \boldsymbol{\mu_j} \in \mathbb{R}^d$, $\|\boldsymbol{x_i}\|_2 = \|\boldsymbol{\mu_j}\|_2 = 1$

$$g_d(\boldsymbol{x_i}|\Theta_C) = \sum_{j=1}^{C} \pi_j \; V_d(\boldsymbol{x_i}|\boldsymbol{\mu_j}, \kappa_j) \qquad \Theta_C = \{(\pi_j, \boldsymbol{\mu_j}, \kappa_j)\}_{1 \leq j \leq C}$$

- Posterior probability $p_{i,j}$ that $\boldsymbol{x_i}$ belongs to identity j :

$$p_{i,j} = \frac{V_d(\boldsymbol{x_i}|\boldsymbol{\mu_j}, \kappa_j)}{\sum_{k=1}^{C} V_d(\boldsymbol{x_i}|\boldsymbol{\mu_k}, \kappa_k)} = \frac{C_d(\kappa_j)\; e^{\kappa_j \; \boldsymbol{\mu_j}^{\mathsf{T}} \boldsymbol{x_i}}}{\sum_{k=1}^{C} C_d(\kappa_k)\; e^{\kappa_k \; \boldsymbol{\mu_k}^{\mathsf{T}} \boldsymbol{x_i}}}$$

Set a statistical model on face embeddings

- Mixture model with C classes : $\boldsymbol{x_i}, \boldsymbol{\mu_j} \in \mathbb{R}^d$, $\|\boldsymbol{x_i}\|_2 = \|\boldsymbol{\mu_j}\|_2 = 1$

$$g_d(\boldsymbol{x_i}|\Theta_C) = \sum_{j=1}^{C} \pi_j \; V_d(\boldsymbol{x_i}|\boldsymbol{\mu_j}, \kappa_j) \qquad \Theta_C = \{(\pi_j, \boldsymbol{\mu_j}, \kappa_j)\}_{1 \leq j \leq C}$$

- Posterior probability $p_{i,j}$ that $\boldsymbol{x_i}$ belongs to identity j :

$$p_{i,j} = \frac{V_d(\boldsymbol{x_i}|\boldsymbol{\mu_j}, \kappa_j)}{\sum_{k=1}^{C} V_d(\boldsymbol{x_i}|\boldsymbol{\mu_k}, \kappa_k)} = \frac{C_d(\kappa_j) \; e^{\kappa_j \; \boldsymbol{\mu_j}^{\intercal} \boldsymbol{x_i}}}{\sum_{k=1}^{C} C_d(\kappa_k) \; e^{\kappa_k \; \boldsymbol{\mu_k}^{\intercal} \boldsymbol{x_i}}}$$

- vMF mixture loss (cross-entropy) :

$$\mathcal{L}_{\text{vMF}}(\{\kappa_j, \boldsymbol{\mu_j}\}_j) = -\frac{1}{N} \sum_{i=1}^{N} \log \left[\frac{C_d(\kappa_{y_i}) \; e^{\kappa_{y_i} \; \boldsymbol{\mu_{y_i}}^{\intercal} \boldsymbol{x_i}}}{\sum_{k=1}^{C} C_d(\kappa_k) \; e^{\kappa_k \; \boldsymbol{\mu_k}^{\intercal} \boldsymbol{x_i}}} \right]$$

Set a statistical model on face embeddings

- vMF mixture loss (cross-entropy) :

$$\mathcal{L}_{\text{vMF}}(\{\kappa_j, \boldsymbol{\mu_j}\}_j) = -\frac{1}{N}\sum_{i=1}^{N} \log\left[\frac{C_d(\kappa_{y_i})\ e^{\kappa_{y_i}\ \boldsymbol{\mu_{y_i}}^\top \boldsymbol{x_i}}}{\sum_{k=1}^{C} C_d(\kappa_k)\ e^{\kappa_k\ \boldsymbol{\mu_k}^\top \boldsymbol{x_i}}}\right]$$

$\rightarrow$ κ_j are hyperparameters

$\rightarrow$ Use $\kappa_{\text{F}}, \kappa_{\text{M}}$

Set a statistical model on face embeddings

- vMF mixture loss (cross-entropy) :

$$\mathcal{L}_{\text{vMF}}(\{\kappa_j, \boldsymbol{\mu_j}\}_j) = -\frac{1}{N}\sum_{i=1}^{N} \log \left[\frac{C_d(\kappa_{y_i})\ e^{\kappa_{y_i}\ \boldsymbol{\mu_{y_i}}^{\mathsf{T}} \boldsymbol{x_i}}}{\sum_{k=1}^{C} C_d(\kappa_k)\ e^{\kappa_k\ \boldsymbol{\mu_k}^{\mathsf{T}} \boldsymbol{x_i}}} \right]$$

$\longrightarrow \kappa_j$ are hyperparameters

$\longrightarrow$ Use $\kappa_{\text{F}}, \kappa_{\text{M}}$

- Generalization of the modified softmax loss :

$$\mathcal{L}_{\text{modified}} = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{\kappa\ \boldsymbol{\mu_{y_i}}^{\mathsf{T}} \boldsymbol{x_i}}}{\sum_{k=1}^{C} e^{\kappa\ \boldsymbol{\mu_k}^{\mathsf{T}} \boldsymbol{x_i}}}$$

Visualization after training

Training data : $N = 400$ images, $C = 20$ IDs, 50% F, 50% M

All parameters from toy model MLP + $(\{\boldsymbol{\mu_j}\})_{1 \leq j \leq C}$ learned

$\kappa_j = \kappa$ fixed for $j = 1 \ldots C$, $\qquad d = 3$

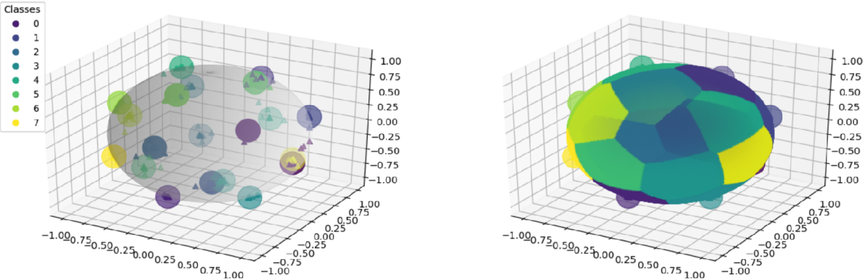


Figure – Visualization of hypersphere after a few training epochs.

Visualization after training

Training data : $N = 400$ images, $C = 20$ identities, 50% F, 50% M

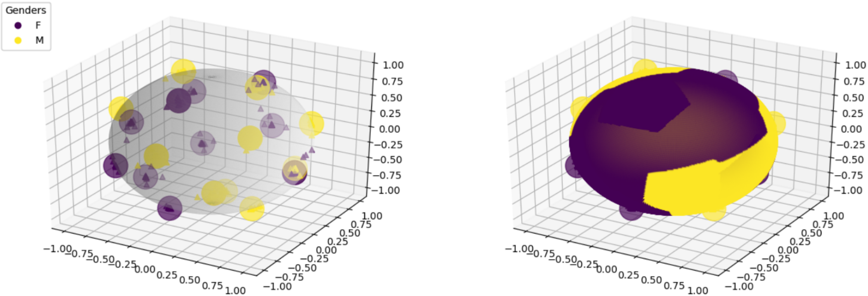
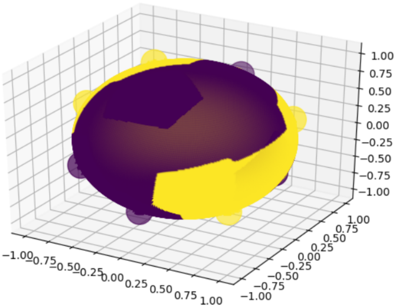


Figure – Visualization of hypersphere after a few training epochs.

Visualization after training

Training data : $N = 400$ images, $C = 20$ identities, 50% F, 50% M



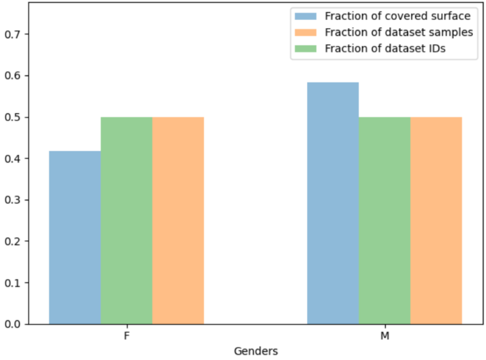


Figure – Hypersphere and covered surface by each gender after a few training epochs.

Data

Datasets	Images	Identities	Images/identity	Gender prop.
Training sets	120k	20k	6	γ
Validation sets	20k	20k	1	γ
Testing set	7k	1k	7	$\gamma_{\text{test}} = 0.5$

Table – Datasets used to train, monitor and test our proposed ethical module. For a given dataset, each identity has exactly the same number of face images.

All parameters from ethical module + $(\{\boldsymbol{\mu_j}\})_{1 \leq j \leq C}$ learned

$d = 256$

Performance of trained model

$\gamma = 0.5$, $\kappa = 10$, 100 epochs

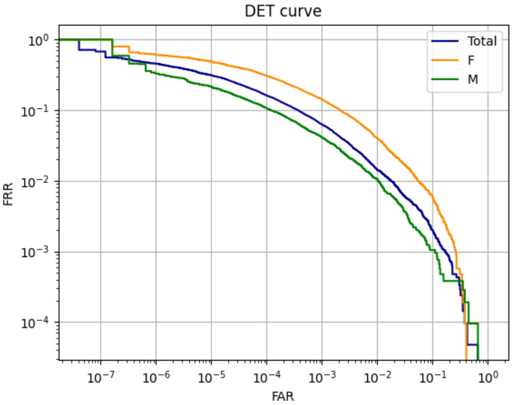


Figure – DET curve of a model trained on gender-balanced training set.

Performance baseline

Baseline = without 'ethical training'

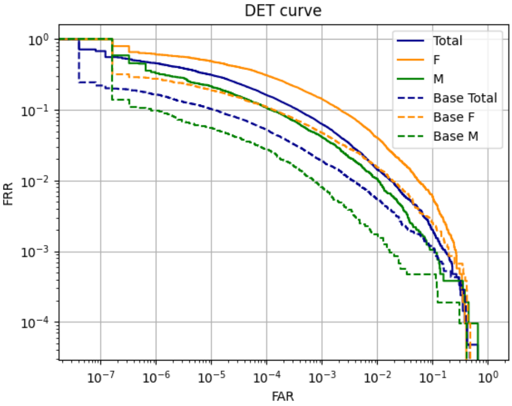


Figure – DET curve of a model trained on gender-balanced training set.

Gender bias from dataset

$\kappa = 10$, 100 epochs

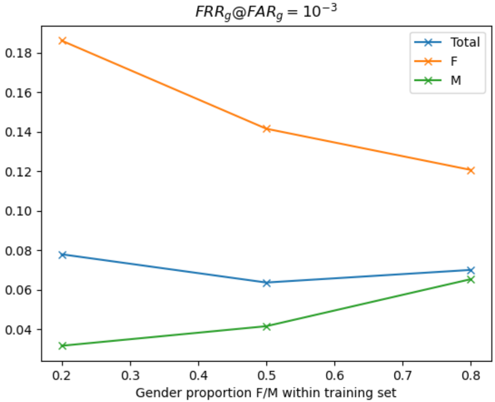


Figure – $\mathrm{FRR}_g@\mathrm{FAR}_g = 10^{-3}$ as function of γ, w.r.t. test data.

2 concentration parameters

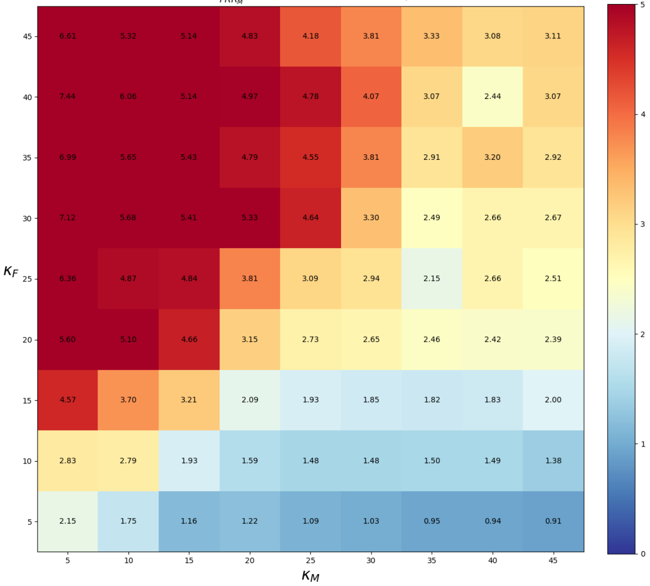
- Gender of k-th identity : $g_k \in \{\mathrm{F}, \mathrm{M}\}$
- vMF loss for gender bias reduction :

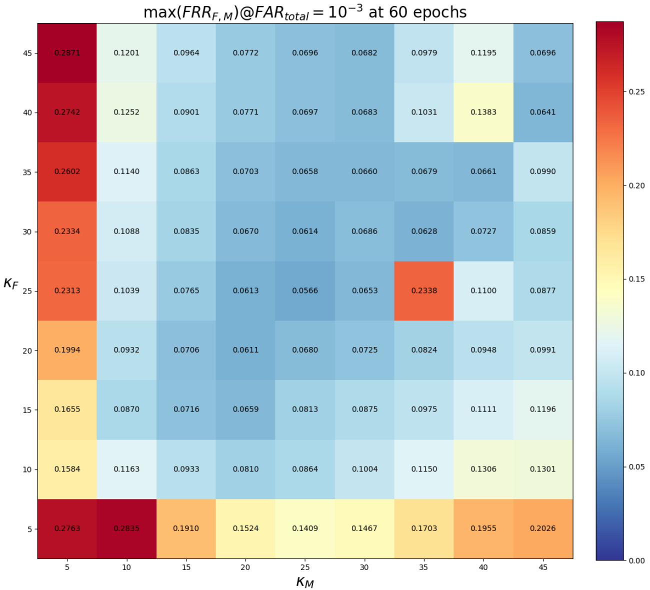
$$\mathcal{L}_{\text{vMF\_gender}}(\Theta_C) = -\frac{1}{N}\sum_{i=1}^{N} \log \left[\frac{C_d(\kappa_{g_{y_i}})\ e^{\kappa_{g_{y_i}}\ \boldsymbol{\mu_{y_i}}^{\mathsf{T}} \boldsymbol{x_i}}}{\sum_{k=1}^{C} C_d(\kappa_{g_k})\ e^{\kappa_{g_k}\ \boldsymbol{\mu_k}^{\mathsf{T}} \boldsymbol{x_i}}} \right]$$

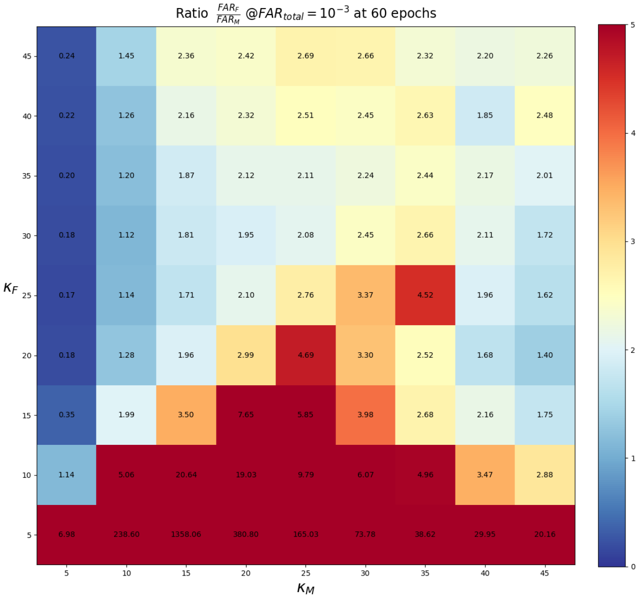
$\rightarrow \kappa_F,\ \kappa_M$

- Biased dataset : $\gamma = 0.2$

Ratio $\frac{FRR_F}{FRR_M}$ @$FAR_{total} = 10^{-3}$ at 60 epochs




max($FRR_{F,M}$)@$FAR_{total} = 10^{-3}$ at 60 epochs
K_F
K_M
45: 0.2871 0.1201 0.0964 0.0772 0.0696 0.0682 0.0979 0.1195 0.0696
40: 0.2742 0.1252 0.0901 0.0771 0.0697 0.0683 0.1031 0.1383 0.0641
35: 0.2602 0.1140 0.0863 0.0703 0.0658 0.0660 0.0679 0.0661 0.0990
30: 0.2334 0.1088 0.0835 0.0670 0.0614 0.0686 0.0628 0.0727 0.0859
25: 0.2313 0.1039 0.0765 0.0613 0.0566 0.0653 0.2338 0.1100 0.0877
20: 0.1994 0.0932 0.0706 0.0611 0.0680 0.0725 0.0824 0.0948 0.0991
15: 0.1655 0.0870 0.0716 0.0659 0.0813 0.0875 0.0975 0.1111 0.1196
10: 0.1584 0.1163 0.0933 0.0810 0.0864 0.1004 0.1150 0.1306 0.1301
5: 0.2763 0.2835 0.1910 0.1524 0.1409 0.1467 0.1703 0.1955 0.2026
5 10 15 20 25 30 35 40 45
0.00 0.05 0.10 0.15 0.20 0.25



Ratio $\frac{FAR_F}{FAR_M}$ @$FAR_{total} = 10^{-3}$ at 60 epochs
K_F
45 0.24 1.45 2.36 2.42 2.69 2.66 2.32 2.20 2.26
40 0.22 1.26 2.16 2.32 2.51 2.45 2.63 1.85 2.48
35 0.20 1.20 1.87 2.12 2.11 2.24 2.44 2.17 2.01
30 0.18 1.12 1.81 1.95 2.08 2.45 2.66 2.11 1.72
25 0.17 1.14 1.71 2.10 2.76 3.37 4.52 1.96 1.62
20 0.18 1.28 1.96 2.99 4.69 3.30 2.52 1.68 1.40
15 0.35 1.99 3.50 7.65 5.85 3.98 2.68 2.16 1.75
10 1.14 5.06 20.64 19.03 9.79 6.07 4.96 3.47 2.88
5 6.98 238.60 1358.06 380.80 165.03 73.78 38.62 29.95 20.16
5 10 15 20 25 30 35 40 45
K_M
0 1 2 3 4 5

Conclusion

- Gender bias reduction, in terms of both FAR and FRR, for any pre-trained model
- To reach performance baseline, train with more data (nb of images fixed / ID ?)
- Link between gender bias and proportion of covered surface by each gender on hypersphere ?
- Can be adapted for more than 2 groups.

References

vmf

crystal_loss

normface

nist_report_2020

References

det_example

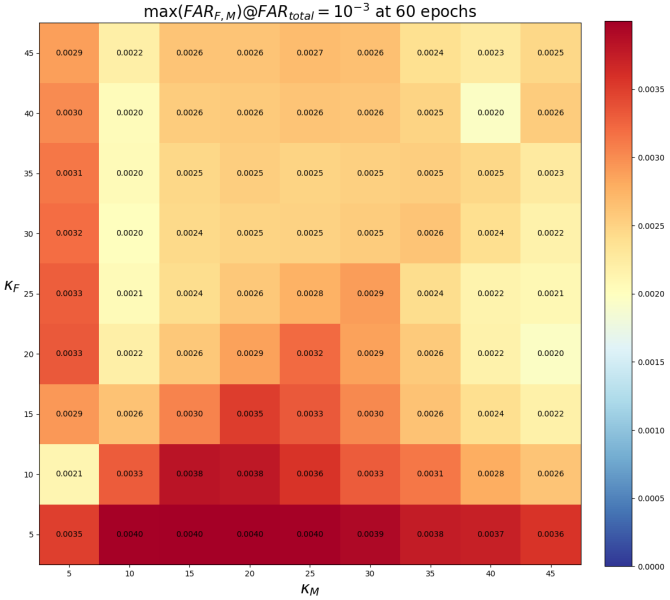
vggface2

agenda_gan_gender

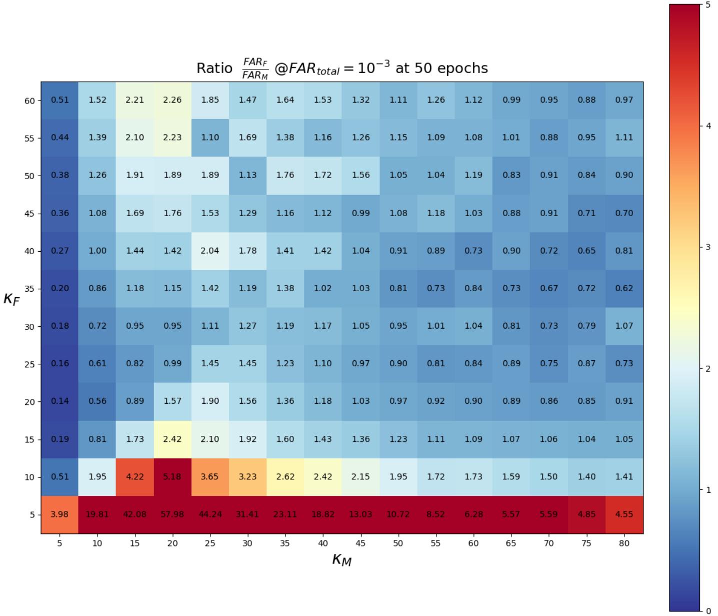
References

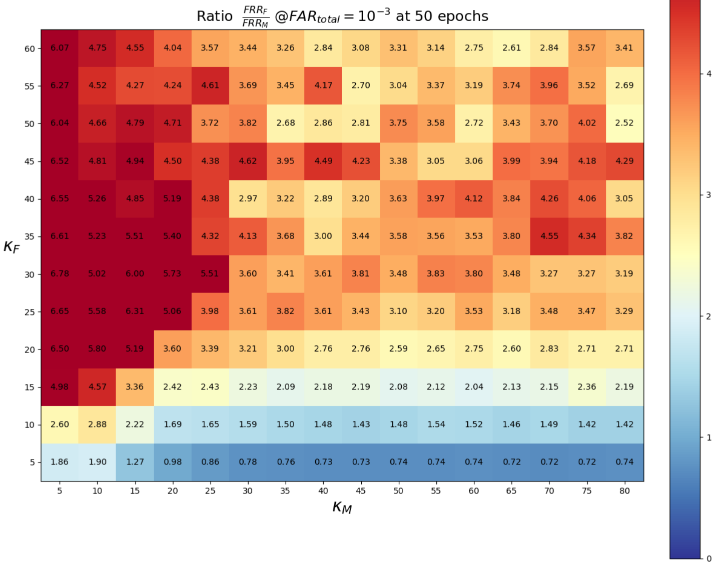
comparison_level_race_bias

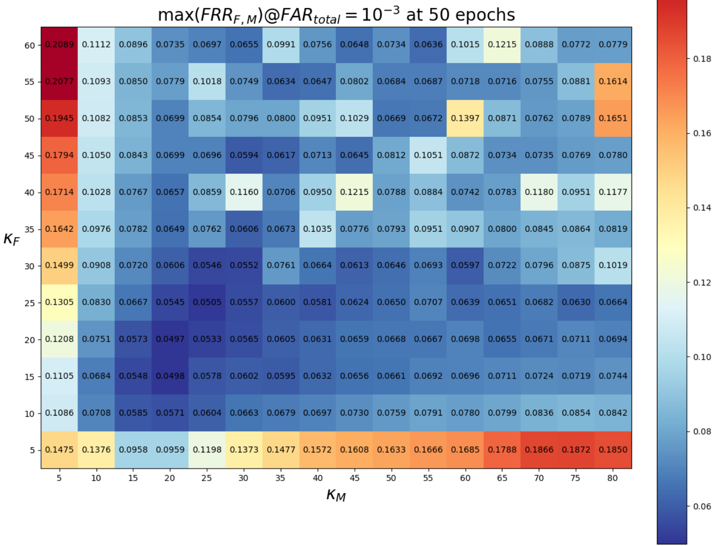
nist_prez_demograpics

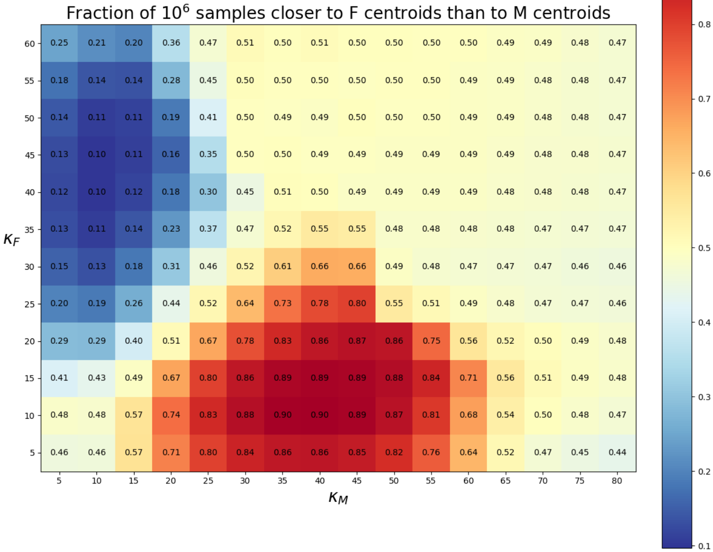


$\max(FAR_{F,M})@FAR_{total} = 10^{-3}$ at 60 epochs
κ_F
κ_M
45: 0.0029 0.0022 0.0026 0.0026 0.0027 0.0026 0.0024 0.0023 0.0025
40: 0.0030 0.0020 0.0026 0.0026 0.0026 0.0026 0.0025 0.0020 0.0026
35: 0.0031 0.0020 0.0025 0.0025 0.0025 0.0025 0.0025 0.0025 0.0023
30: 0.0032 0.0020 0.0024 0.0025 0.0025 0.0025 0.0026 0.0024 0.0022
25: 0.0033 0.0021 0.0024 0.0026 0.0028 0.0029 0.0024 0.0022 0.0021
20: 0.0033 0.0022 0.0026 0.0029 0.0032 0.0029 0.0026 0.0022 0.0020
15: 0.0029 0.0026 0.0030 0.0035 0.0033 0.0030 0.0026 0.0024 0.0022
10: 0.0021 0.0033 0.0038 0.0038 0.0036 0.0033 0.0031 0.0028 0.0026
5: 0.0035 0.0040 0.0040 0.0040 0.0040 0.0039 0.0038 0.0037 0.0036
5 10 15 20 25 30 35 40 45
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000


Ratio $\frac{FAR_F}{FAR_M}$ @$FAR_{total} = 10^{-3}$ at 50 epochs
K_F
K_M
60: 0.51 1.52 2.21 2.26 1.85 1.47 1.64 1.53 1.32 1.11 1.26 1.12 0.99 0.95 0.88 0.97
55: 0.44 1.39 2.10 2.23 1.10 1.69 1.38 1.16 1.26 1.15 1.09 1.08 1.01 0.88 0.95 1.11
50: 0.38 1.26 1.91 1.89 1.89 1.13 1.76 1.72 1.56 1.05 1.04 1.19 0.83 0.91 0.84 0.90
45: 0.36 1.08 1.69 1.76 1.53 1.29 1.16 1.12 0.99 1.08 1.18 1.03 0.88 0.91 0.71 0.70
40: 0.27 1.00 1.44 1.42 2.04 1.78 1.41 1.42 1.04 0.91 0.89 0.73 0.90 0.72 0.65 0.81
35: 0.20 0.86 1.18 1.15 1.42 1.19 1.38 1.02 1.03 0.81 0.73 0.84 0.73 0.67 0.72 0.62
30: 0.18 0.72 0.95 0.95 1.11 1.27 1.19 1.17 1.05 0.95 1.01 1.04 0.81 0.73 0.79 1.07
25: 0.16 0.61 0.82 0.99 1.45 1.45 1.23 1.10 0.97 0.90 0.81 0.84 0.89 0.75 0.87 0.73
20: 0.14 0.56 0.89 1.57 1.90 1.56 1.36 1.18 1.03 0.97 0.92 0.90 0.89 0.86 0.85 0.91
15: 0.19 0.81 1.73 2.42 2.10 1.92 1.60 1.43 1.36 1.23 1.11 1.09 1.07 1.06 1.04 1.05
10: 0.51 1.95 4.22 5.18 3.65 3.23 2.62 2.42 2.15 1.95 1.72 1.73 1.59 1.50 1.40 1.41
5: 3.98 19.81 42.08 57.98 44.24 31.41 23.11 18.82 13.03 10.72 8.52 6.28 5.57 5.59 4.85 4.55
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80
0 1 2 3 4 5


Ratio $\frac{FRR_F}{FRR_M}$ @$FAR_{total} = 10^{-3}$ at 50 epochs
K_F
K_M
60: 6.07 4.75 4.55 4.04 3.57 3.44 3.26 2.84 3.08 3.31 3.14 2.75 2.61 2.84 3.57 3.41
55: 6.27 4.52 4.27 4.24 4.61 3.69 3.45 4.17 2.70 3.04 3.37 3.19 3.74 3.96 3.52 2.69
50: 6.04 4.66 4.79 4.71 3.72 3.82 2.68 2.86 2.81 3.75 3.58 2.72 3.43 3.70 4.02 2.52
45: 6.52 4.81 4.94 4.50 4.38 4.62 3.95 4.49 4.23 3.38 3.05 3.06 3.99 3.94 4.18 4.29
40: 6.55 5.26 4.85 5.19 4.38 2.97 3.22 2.89 3.20 3.63 3.97 4.12 3.84 4.26 4.06 3.05
35: 6.61 5.23 5.51 5.40 4.32 4.13 3.68 3.00 3.44 3.58 3.56 3.53 3.80 4.55 4.34 3.82
30: 6.78 5.02 6.00 5.73 5.51 3.60 3.41 3.61 3.81 3.48 3.83 3.80 3.48 3.27 3.27 3.19
25: 6.65 5.58 6.31 5.06 3.98 3.61 3.82 3.61 3.43 3.10 3.20 3.53 3.18 3.48 3.47 3.29
20: 6.50 5.80 5.19 3.60 3.39 3.21 3.00 2.76 2.76 2.59 2.65 2.75 2.60 2.83 2.71 2.71
15: 4.98 4.57 3.36 2.42 2.43 2.23 2.09 2.18 2.19 2.08 2.12 2.04 2.13 2.15 2.36 2.19
10: 2.60 2.88 2.22 1.69 1.65 1.59 1.50 1.48 1.43 1.48 1.54 1.52 1.46 1.49 1.42 1.42
5: 1.86 1.90 1.27 0.98 0.86 0.78 0.76 0.73 0.73 0.74 0.74 0.74 0.72 0.72 0.72 0.74
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80
0 1 2 3 4 5


max($FRR_{F,M}$)@FAR_{total} = 10^{-3} at 50 epochs
60 0.2089 0.1112 0.0896 0.0735 0.0697 0.0655 0.0991 0.0756 0.0648 0.0734 0.0636 0.1015 0.1215 0.0888 0.0772 0.0779
55 0.2077 0.1093 0.0850 0.0779 0.1018 0.0749 0.0634 0.0647 0.0802 0.0684 0.0687 0.0718 0.0716 0.0755 0.0881 0.1614
50 0.1945 0.1082 0.0853 0.0699 0.0854 0.0796 0.0800 0.0951 0.1029 0.0669 0.0672 0.1397 0.0871 0.0762 0.0789 0.1651
45 0.1794 0.1050 0.0843 0.0699 0.0696 0.0594 0.0617 0.0713 0.0645 0.0812 0.1051 0.0872 0.0734 0.0735 0.0769 0.0780
40 0.1714 0.1028 0.0767 0.0657 0.0859 0.1160 0.0706 0.0950 0.1215 0.0788 0.0884 0.0742 0.0783 0.1180 0.0951 0.1177
35 0.1642 0.0976 0.0782 0.0649 0.0762 0.0606 0.0673 0.1035 0.0776 0.0793 0.0951 0.0907 0.0800 0.0845 0.0864 0.0819
30 0.1499 0.0908 0.0720 0.0606 0.0546 0.0552 0.0761 0.0664 0.0613 0.0646 0.0693 0.0597 0.0722 0.0796 0.0875 0.1019
25 0.1305 0.0830 0.0667 0.0545 0.0505 0.0557 0.0600 0.0581 0.0624 0.0650 0.0707 0.0639 0.0651 0.0682 0.0630 0.0664
20 0.1208 0.0751 0.0573 0.0497 0.0533 0.0565 0.0605 0.0631 0.0659 0.0668 0.0667 0.0698 0.0655 0.0671 0.0711 0.0694
15 0.1105 0.0684 0.0548 0.0498 0.0578 0.0602 0.0595 0.0632 0.0656 0.0661 0.0692 0.0696 0.0711 0.0724 0.0719 0.0744
10 0.1086 0.0708 0.0585 0.0571 0.0604 0.0663 0.0679 0.0697 0.0730 0.0759 0.0791 0.0780 0.0799 0.0836 0.0854 0.0842
5 0.1475 0.1376 0.0958 0.0959 0.1198 0.1373 0.1477 0.1572 0.1608 0.1633 0.1666 0.1685 0.1788 0.1866 0.1872 0.1850
K_F
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80
K_M
0.20 0.18 0.16 0.14 0.12 0.10 0.08 0.06


Fraction of 10^6 samples closer to F centroids than to M centroids
κ_F
κ_M
60 0.25 0.21 0.20 0.36 0.47 0.51 0.50 0.51 0.50 0.50 0.50 0.50 0.49 0.49 0.48 0.47
55 0.18 0.14 0.14 0.28 0.45 0.50 0.50 0.50 0.50 0.50 0.50 0.49 0.49 0.48 0.48 0.47
50 0.14 0.11 0.11 0.19 0.41 0.50 0.49 0.49 0.50 0.50 0.50 0.49 0.49 0.48 0.48 0.47
45 0.13 0.10 0.11 0.16 0.35 0.50 0.50 0.49 0.49 0.49 0.49 0.49 0.49 0.48 0.48 0.47
40 0.12 0.10 0.12 0.18 0.30 0.45 0.51 0.50 0.49 0.49 0.49 0.49 0.48 0.48 0.48 0.47
35 0.13 0.11 0.14 0.23 0.37 0.47 0.52 0.55 0.55 0.48 0.48 0.48 0.48 0.47 0.47 0.47
30 0.15 0.13 0.18 0.31 0.46 0.52 0.61 0.66 0.66 0.49 0.48 0.47 0.47 0.47 0.46 0.46
25 0.20 0.19 0.26 0.44 0.52 0.64 0.73 0.78 0.80 0.55 0.51 0.49 0.48 0.47 0.47 0.46
20 0.29 0.29 0.40 0.51 0.67 0.78 0.83 0.86 0.87 0.86 0.75 0.56 0.52 0.50 0.49 0.48
15 0.41 0.43 0.49 0.67 0.80 0.86 0.89 0.89 0.89 0.88 0.84 0.71 0.56 0.51 0.49 0.48
10 0.48 0.48 0.57 0.74 0.83 0.88 0.90 0.90 0.89 0.87 0.81 0.68 0.54 0.50 0.48 0.47
5 0.46 0.46 0.57 0.71 0.80 0.84 0.86 0.86 0.85 0.82 0.76 0.64 0.52 0.47 0.45 0.44
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80
0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1