THALES
Building a future we can all trust

Méthodes in-processing de mitigation des biais

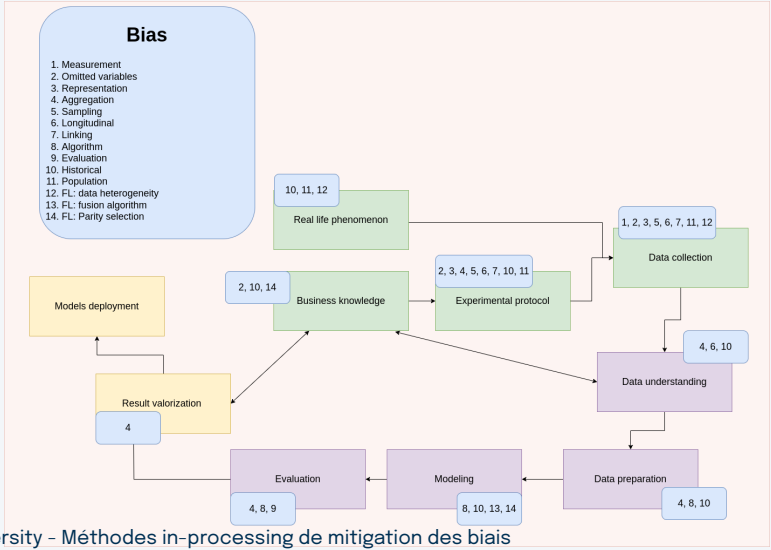
Session 5:
Mitiger les biais avec des méthodes de type in-processing

Alice HELIOU and Vincent THOUVENOT
Laboratoire de Data Science de CortAIx Labs

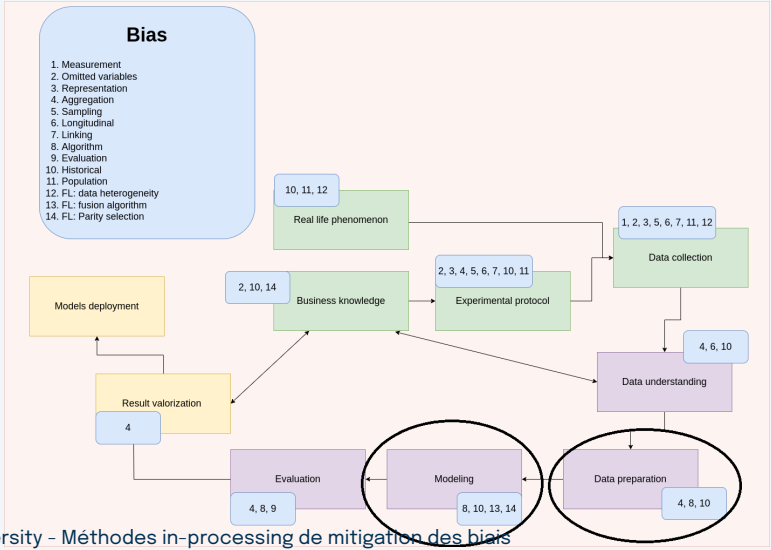
Content

Introduction

Position du cours



Position du cours



Juste retirer l'attribut sensible?

Example: une banque cherche une méthode pour savoir à qui faire un prêt de manière juste

- Objectif de la banque: déterminer qui va rembourser le prêt;
- Le prêt est destiné à deux groupes disjoints de personnes;
- Juste retirer du modèle l'information à propos de l'attribut sensible peut être inefficace à cause des interactions indirectes avec d'autres attributs.

Juste retirer l'attribut sensible n'est pas la solution optimale: les garder peut être plus efficace !

Méthodes de correction des biais

- **Pre-processing:** Modification des bases de données d'apprentissage
- **In-processing:** Correction des biais apprise en même temps que le modèle de machine learning
- **Post-processing:** Modification des prédictions des modèles de machine learning

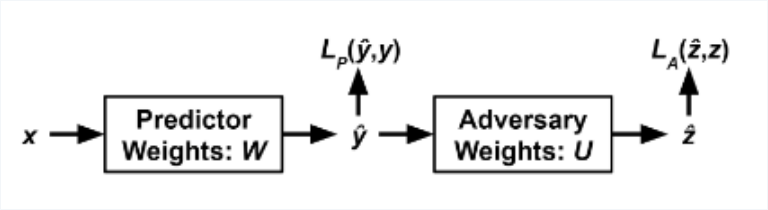
Méthodes de correction des biais

- **Pre-processing:** Modification des bases de données d'apprentissage
- **In-processing: Correction des biais apprise en même temps que le modèle de machine learning**
- **Post-processing:** Modification des prédictions des modèles de machine learning

Méthodes d’AIF360

Adversarial debiaising

B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating Unwanted Biases with Adversarial Learning," AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society, 2018.



Prejudice Remover

Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-aware Classifier with Prejudice Remover Regularizer. Machine Learning and Knowledge Discovery in Databases (2012), 35–50.

$$-\mathcal{L}(\mathcal{D};\boldsymbol{\Theta}) + \eta \mathrm{R}(\mathcal{D},\boldsymbol{\Theta}) + \frac{\lambda}{2}\|\boldsymbol{\Theta}\|_2^2,$$

Prejudice Remover

Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-aware Classifier with Prejudice Remover Regularizer. Machine Learning and Knowledge Discovery in Databases (2012), 35–50.

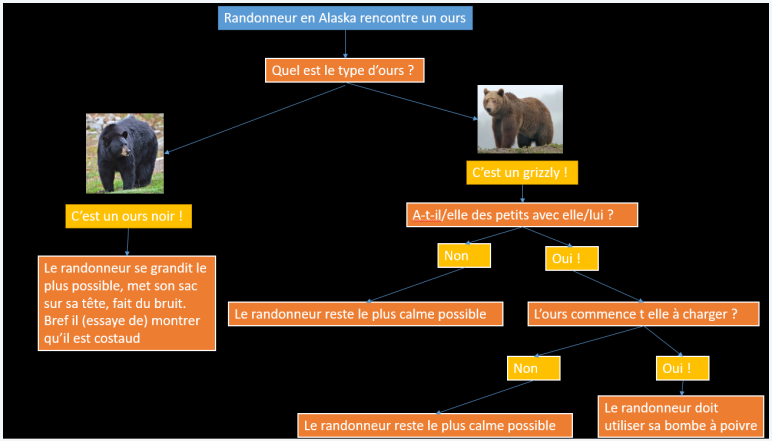
$$\sum_{(\mathbf{x}_i, s_i) \in \mathcal{D}} \sum_{y \in \{0,1\}} \mathcal{M}[y|\mathbf{x}_i, s_i; \boldsymbol{\Theta}] \ln \frac{\hat{\Pr}[y|s_i]}{\hat{\Pr}[y]},$$

Exemple d’adaptation d’un algorithme de machine learning: cas des forêts aléatoires

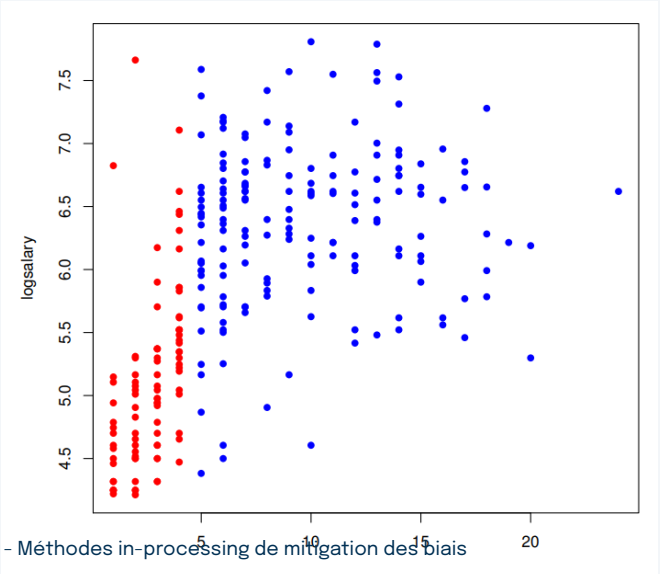
Introduction aux arbres de décision

- Arbre de décision: outils utilisant un arbre comme graphe de décision et leurs conséquences possibles
- Créer un ensemble de feuilles (constituant des décisions) et de branches (expliquant comment on fait pour arriver à la décision
  - Par exemple, si la température est inférieure à 8 degrés Celsius et qu'il est 5h du matin (branches), alors la consommation électrique prédite est de 800 MWh (feuille)
- La structure en feuilles et en branches mime la forme d'un arbre
- Peut être utilisé pour de la régression ou de la classification

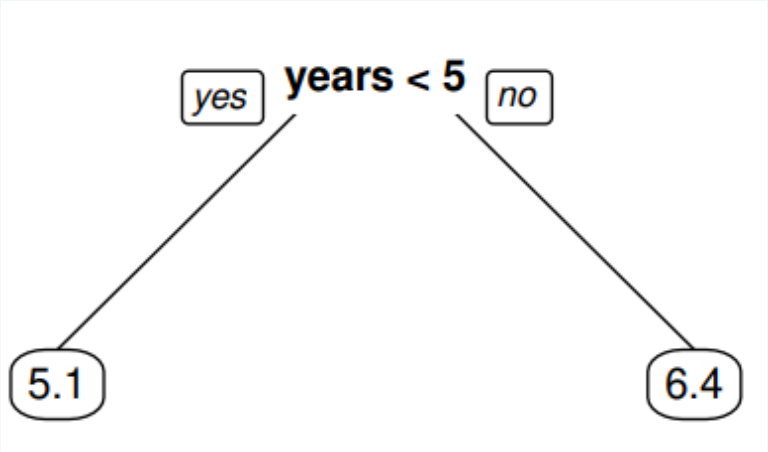
Raisonnement binaire: rencontre entre un randonneur et un ours en Alaska



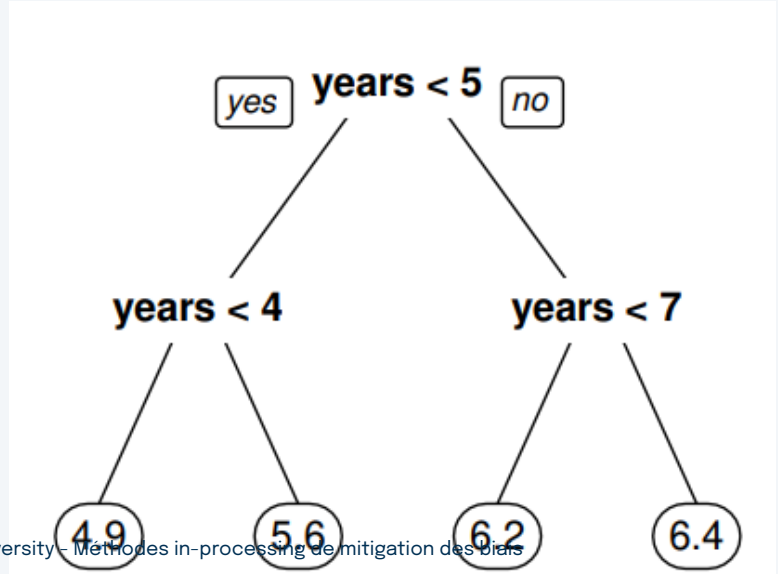
Intuition



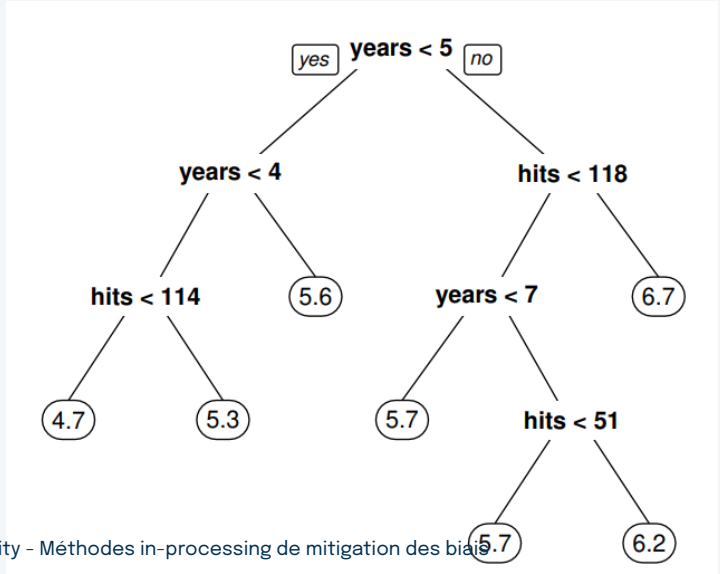
Intuition



Intuition



Intuition


yes
years < 5
no
years < 4
hits < 118
hits < 114
5.6
years < 7
6.7
4.7
5.3
5.7
hits < 51
5.7
6.2

Construction des arbres

- La racine de notre arbre regroupe tous les points de notre jeu de données
- À la première itération, l'idée va être de considérer toutes les valeurs possibles de seuil t et des variables j qui sépareraient notre jeu de données en deux sous-groupes
- On choisit le seuil et la variable qui permet des minimiser la MSE (Mean Square Error) pondérée des deux sous groupes
- On réitère le procédé pour chaque sous groupe

Régle d'arrêt

Le principe de l'algorithme CART est de ne pas fixer de règle d'arrêt arbitraire pour la procédure.
L'algorithme s'arrête de diviser les feuilles quand:

- Il n'y a qu'une observation dans la feuille ou
- Les individus de la feuille ont les mêmes valeurs

On construit ainsi l'arbre maximal que l'on élaguera ensuite.

Elagage des arbres

- Nous partons de l'arbre maximal
- On sélectionne une suite d'arbres emboités. La sélection s'effectue en optimisant un critère Cout/complexité qui permet de réguler le compromis entre ajustement et complexité de l'arbre.
- On sélectionne un arbre dans cette sous-suite en optimisant un critère de performance.

Avantage des arbres

- Facile à interpréter
- Fonctionne à la fois pour la classification et la régression
- Prend en compte les variables catégorielles naturellement
- Pas d'hypothèse formelle sur la distribution des données
- Peut modéliser des fortes non linéarités
- Peut gérer les valeurs manquantes

Inconvénients des arbres

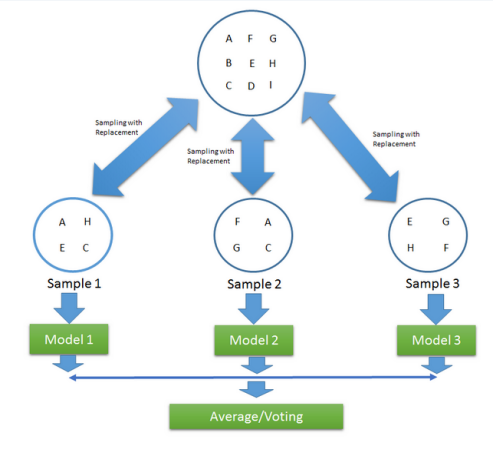
- Si la variable sélectionnée pour la première division change, l'arbre va complètement changer
  - Possible par exemple si l'arbre est construit sur deux échantillons différents
- De ce fait, les arbres sont peu robustes
- On préfera des méthodes qui utilisent des ensembles d'arbres, la diversité des arbres améliorant la robustesse

Intuition des Forêts Aléatoires

- Forêt: ensemble d'arbres
- Aléatoire: construction de plusieurs sous-arbres différents
- Les forêts aléatoires consistent à faire tourner en parallèle plusieurs arbres de décisions construits aléatoirement, avant de les moyenner.
- Si les arbres sont décorrélés, cela permet de réduire la variance des prédictions

Bagging

- Echantillon boostrap: échantillonnage d'un jeu de données avec un tirage uniforme avec remise
- Bagging:
  - Créer *M* échantillons bootstrap
  - Apprendre sur chaque échantillon un modèle
  - Agréger les prédictions des modèles

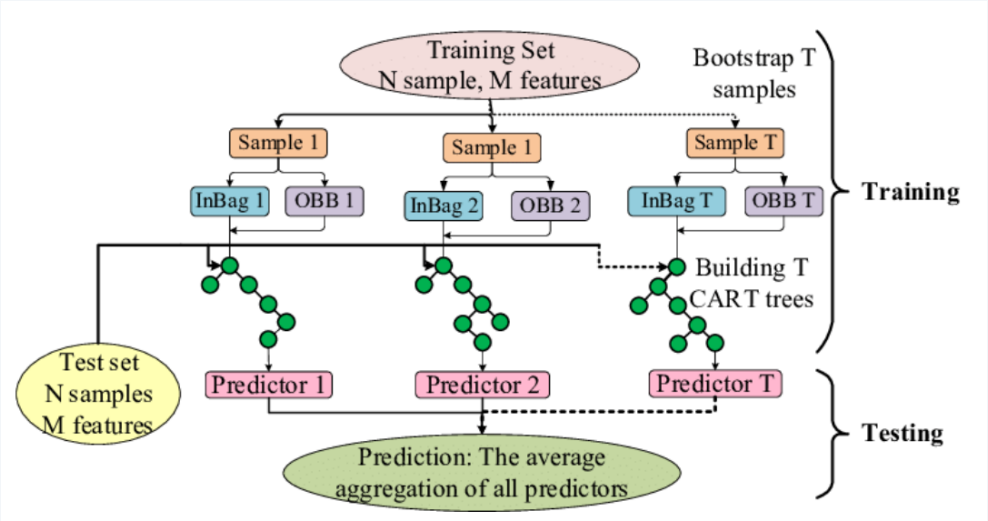


Algorithme des Forets Aléatoires

1. For $b \in \{1, \ldots, B\}$:
   - 1.1 Draw a bootstrap sample $Z*$ of size N from the training data;
   - 1.2 Grow a RF tree T_b to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size n_{min} is reached:
     - 1.2.1 Select m variables at random from the p variables;
     - 1.2.2 Pick the best feature/variable/split-point among the m;
     - 1.2.3 Split the node into two daughter nodes.
2. Output the ensemble of trees $\{T_b\}_1^B$.

To make a prediction at a new point x:
$\hat{f}_{rf}^B = \frac{1}{B} \sum_{b=1}^{B} T_b(x)$.

Echantillon “Out-Of-Bag”



MDI and MDA, deux critères d'importance des variables

- Objectif: classer les variables selon leur importance sur les sorties/les erreurs du modèle
- Mean Decreasing Impurity (MDI): A chaque division de l'arbre où la variable est impliquée, l'amélioration du critère de division est calculée et accumulée sur tous les arbres. Si le MDI est élevée, la variable est importante
- Mean Decreasing Accuracy (MDA): Comparaison entre les erreurs OOB lorsque la variable est inchangée et lorsque celle-ci est aléatoirement permutée. Si le MDA est élevé, alors la variable est importante.
- MDI plus rapide à calculer, mais biaisé (tendance à exagérer l'importance des vairables continues ou avec une forte cardinalité, calculer sur les données d'apprentissage)
- MDA possiblement biaisé en cas de forte multi-collinarité entre les variables explicatives

Fair Random Forest

Fair Forests: Regularized Tree Induction to Minimize Model Bias, Raff, Sylvester et Mills, 2017

$$G(T, a) = I(T) - \sum_{\forall T_i \in \text{splits}(a)} \frac{|T_i|}{|T|} \cdot I(T_i)$$

Fair Random Forest

Fair Forests: Regularized Tree Induction to Minimize Model Bias, Raff, Sylvester et Mills, 2017

$$I_{\text{Gini}}(T) = 1 - \sum_{\forall T_i \in \text{splits(label)}} \left(\frac{|T_i|}{|T|} \right)^2$$

Fair Random Forest

Fair Forests: Regularized Tree Induction to Minimize Model Bias, Raff, Sylvester et Mills, 2017

$$I^a_{\text{Gini}}(T) = \frac{1 - \sum_{\forall T_i \in \text{splits}(a)} \left(\frac{|T_i|}{|T|}\right)^2}{1 - |\text{splits}(a)|^{-1}}$$

Fair Random Forest

Fair Forests: Regularized Tree Induction to Minimize Model Bias, Raff, Sylvester et Mills, 2017

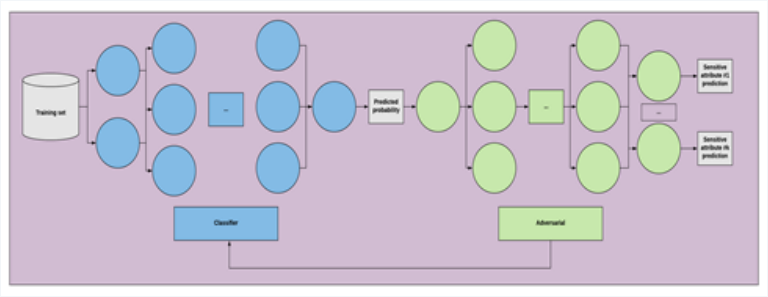
$$G_{\text{fair}}(T, b) = G^{l}(T, b) - G^{af}(T, b)$$

Exemple d’un modèle adversaire

Contexte

- Stage de Ilyes Mahammed Chikouche portant sur les liens entre la Privacy et la Fairness;
- Entre autres choses, étude et implémentation du modèle adversaire pour assurer la fairness.

Fair Adversarial Network



Louppe, Kagan, and Cranmer, Learning to Pivot with Adversarial Networks, 2017

Process

Note:

- $\widehat{\mathbf{y}}$ classifier prediction based on input X, $\mathbf{y}$ the true value and Z the protected attributes
- $\theta_{\mathbf{clf}}$ and $\theta_{\mathbf{ad}}$ parameters of respectively the classifer and the adversarial networks
- $Loss_Y(\theta_{\mathbf{clf}})$ and $Loss_Z(\theta_{\mathbf{clf}}, \theta_{\mathbf{ad}})$ the loss of pre-trained classifier and adversarial networks

During the iteration, the following objective function function are considered:

- Classifier: $\min_{\theta_{\mathbf{clf}}} (Loss_Y(\theta_{\mathbf{clf}}) - \lambda Loss_Z(\theta_{\mathbf{clf}}, \theta_{\mathbf{ad}}))$
- Adversarial: $\min_{\theta_{\mathbf{ad}}} Loss_Z(\theta_{\mathbf{clf}}, \theta_{\mathbf{ad}})$

Louppe, Kagan, and Cranmer, Learning to Pivot with Adversarial Networks, 2017

Algorithm

Algorithm 1 Adversarial training of a classifier f against an adversary r.

Inputs: training data $\{x_i, y_i, z_i\}_{i=1}^{N}$; *Outputs:* $\hat{\theta}_f, \hat{\theta}_r$.

1: **for** $t = 1$ to T **do**
2: **for** $k = 1$ to K **do**
3: Sample minibatch $\{x_m, z_m, s_m = f(x_m; \theta_f)\}_{m=1}^{M}$ of size M;
4: With θ_f fixed, update r by ascending its stochastic gradient $\nabla_{\theta_r} E(\theta_f, \theta_r) :=$

$$\nabla_{\theta_r} \sum_{m=1}^{M} \log p_{\theta_r}(z_m|s_m);$$

5: **end for**
6: Sample minibatch $\{x_m, y_m, z_m, s_m = f(x_m; \theta_f)\}_{m=1}^{M}$ of size M;
7: With θ_r fixed, update f by descending its stochastic gradient $\nabla_{\theta_f} E(\theta_f, \theta_r) :=$

$$\nabla_{\theta_f} \sum_{m=1}^{M} \left[-\log p_{\theta_f}(y_m|x_m) + \log p_{\theta_r}(z_m|s_m) \right],$$

where $p_{\theta_f}(y_m|x_m)$ denotes $\mathbf{1}(y_m = 0)(1 - s_m) + \mathbf{1}(y_m = 1)s_m$;
8: **end for**

Louppe, Kagan, and Cranmer, Learning to Pivot with Adversarial Networks, 2017

Bias amplification by ML for COMPAS Dataset

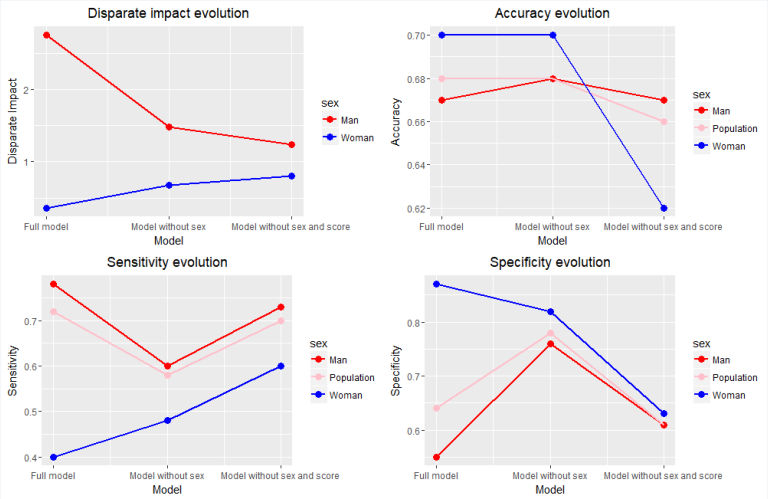
p%-rule on dataset

Gender	age class	Origin
73%	77%	73%

p%-rule of a basic classifier (ROC AUC = 0.76)

Gender	age class	Origin
51%	54%	44%

Remove gender attribute as protection for gender ?



Adversarial Network description

Classifier

- Pre-trained Loss: Binary cross entropy
- Layers:
  - 3 hide dense layers with 3 dropout layers
  - Output layer: sigmoid activation function
- Optimizer: Adam

Adversarial

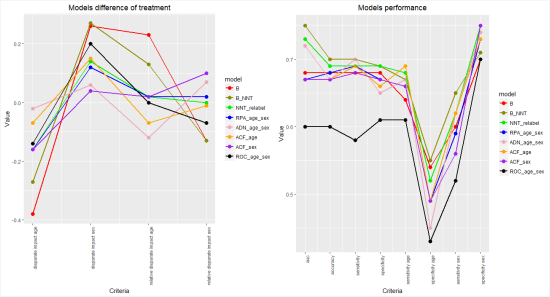
- Loss: Cross entropy
- Layers:
  - 3 hide dense layers
  - Output: sigmoid activation function
- Optimizer: Adam

Protected attributes exclude of features set

Protected attributes include in features set

Benchmarking

- Fairness not considered: Baseline logistic regression (B) and Baseline neural network (B NNT)
- Datasets modification: Neural Network train on relabelized data (NNT relabel) and Logistic regression without protecting attributes (RPA)
- Algorithm modification: **Adversarial network (ADN)** and Additive counterfactually fair model (ACF)
- Algorithm outputs modification: Reject-option classification (ROC)



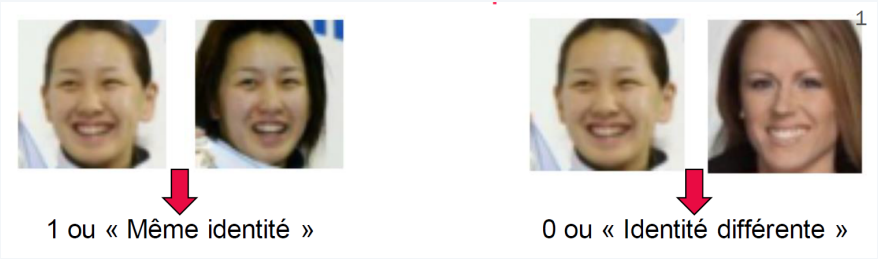
Exemple sur un modèle utilisé en biométrie

Autrice des travaux présentés

- Mélanie Gornet
- Ingénieure ISAE-SUPAERO Sciences des données, MScSciences Po Paris Affaires internationales
- Doctorante Institut Polytechnique de Paris -Télécom Paris,Sciences sociales (droit et régulation).“Ethique, régulation et évaluation des systèmes d’intelligence artificielle”
- Stage CCNE Numérique (anciennement CNPEN) en avril 2021

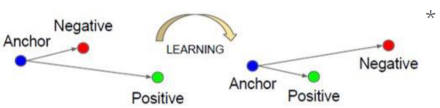
Source: Séminaire ThereSIS, Mélanie Gornet

Reconnaissance faciale par authentification



Source: Séminaire ThereSIS, Mélanie Gornet

Apprentissage par triplets (triplet loss)



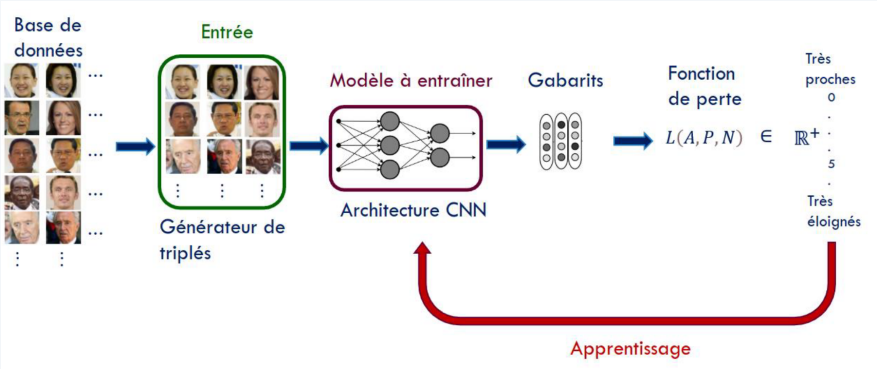
$$\mathrm{d}(A,P) < \mathrm{d}(A,N)$$

$$\alpha + \|f(A) - f(P)\|^2 < \|f(A) - f(N)\|^2$$

$$L(A,P,N) = \max(\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + \alpha, 0)$$

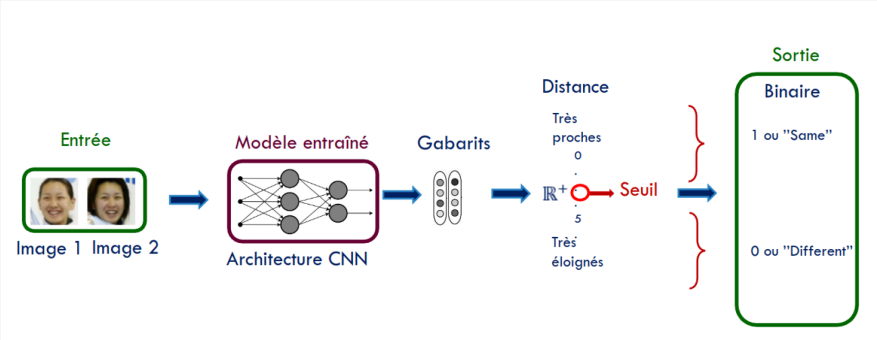
Source: Séminaire ThereSIS, Mélanie Gornet

Fonctionnement global: entraînement



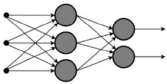
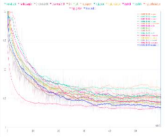
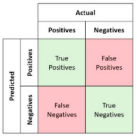
Source: Séminaire ThereSIS, Mélanie Gornet

Fonctionnement global: évaluation



Source: Séminaire ThereSIS, Mélanie Gornet

Choix de conception

Data Acquisition	Data Processing	Neural Network	Training	Evaluation
				
- Image resolution - Number of images - Number of images per person - Labels - Consent - Acquisition technique - Balance - ...	- Normalization - Sampling technique - Augmentation - Number of batches - Train/Test separation - Framing - ...	- CNN - Triplet Loss - Initialization - Layers arrangement - Network depth - ...	- Loss function - Margin - Number of epochs - Hard mining - Regularization - Optimizer - Dropout - Earlystopping - Learning rate and scheduler - Mitigation techniques - ...	- Threshold - Pairs - Metrics - Groups - “Acceptable” value - ...

Source: Séminaire ThereSIS, Mélanie Gornet

Groupes

- Gender: Female vs Male;
- Skin color: Black vs White vs Non-white;
- Intersectional: white male vs non-white female vs black female.

Source: Séminaire ThereSIS, Mélanie Gornet

Métriques

$$TM = \#\{(z_i, z_j) \in G | s(z_i, z_j) < \tau\}$$
$$TNM = \#\{(z_i, z_j) \in I | s(z_i, z_j) \geq \tau\}$$
$$FM = \#\{(z_i, z_j) \in I | s(z_i, z_j) < \tau\}$$
$$FNM = \#\{(z_i, z_j) \in G | s(z_i, z_j) \geq \tau\}$$

- s: similarity measure (distance)
- zi, zj: face embeddings
- τ: decision threshold
- G: set of genuine pairs
- I: set of impostor pairs

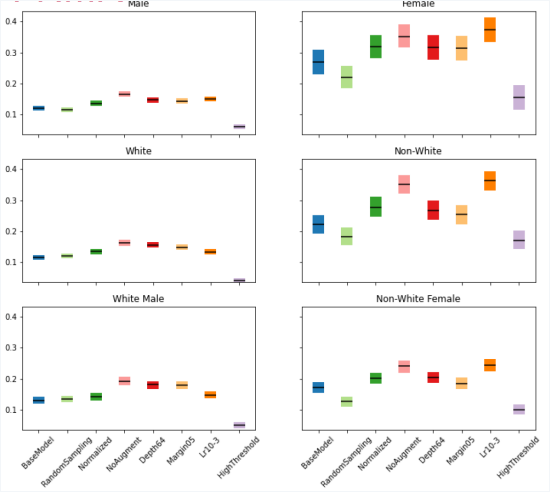
False Match Rate (FMR) – False Non Match Rate (FNMR)

$$FMR = \frac{\#\{(z_i, z_j) \in I | s(z_i, z_j) < \tau\}}{\#\{(z_i, z_j) \in I\}}$$
$$FNMR = \frac{\#\{(z_i, z_j) \in G | s(z_i, z_j) \geq \tau\}}{\#\{(z_i, z_j) \in G\}}$$

$$FMR = \frac{FM}{FM + TNM}$$
$$FNMR = \frac{FNM}{FNM + TM}$$

Source: Séminaire ThereSIS, Mélanie Gornet

FNMR



Source: Séminaire ThereSIS, Mélanie Gornet

Résumé des résultats 1

- **Data sampling:** random sampling better for fairness
- **Data normalisation:** normalization does not improve fairness
- **Data augmentation:** different models yield biases on different metrics/groups
- **Depth of the network:** Depth64 reduces biases but at the cost of overall-performance

Source: Séminaire ThereSIS, Mélanie Gornet

Résumé des résultats 2

Le système le plus “juste” est-il celui qui donne le meilleur score sur les groupes des minorités, ou celui qui réduit l’écart entre les groupes ?

- **Margin:** Margin05 reduces biases but at the cost of overall performance
- **Learning rate:** different models yield biases on different metrics/groups
- **Threshold:** compromise between FMR and FNMR, small improvement of biases with a higher threshold

Source: Séminaire ThereSIS, Mélanie Gornet

Conclusion

- Il n'y a pas que les données qui impactent la fairness!
- L'expérience n'est pas généralisable et doit être refaite pour chaque système;
- La démarche de tester l'impact de différents paramètres "techniques" sur la fairness est généralisable;
- Importance de la responsabilisassions des choix "techniques";
- Repenser la fairness comme une dimension de la performance.

Source: Séminaire ThereSIS, Mélanie Gornet