

Implémentation de DenseWeight et LMFLoss afin de traiter des données déséquilibrées

Simon DURAND

Introduction

Le développement d'algorithmes de machine learning pour l'analyse d'images est un domaine en pleine croissance. Nous utilisons l'ensemble de données UTKFace [1], qui contient des informations sur l'âge, le genre et l'origine ethnique, pour améliorer les performances du modèle MobileNetV3 [2]. Pour gérer le déséquilibre des classes dans les données, nous appliquons deux méthodes, DenseWeight [3] et LMFLoss [4].

Thématique

L'analyse d'images faciales, un domaine clé de la vision par ordinateur avec des applications variées, se heurte souvent à un défi majeur : le traitement des données déséquilibrées. Ce déséquilibre peut entraver l'amélioration des performances des modèles. Ce projet met donc l'accent sur l'optimisation d'un algorithme de machine learning en gérant efficacement le déséquilibre des données, particulièrement en ce qui concerne l'âge et l'origine ethnique.

État de l'art

Pour traiter les données déséquilibrées en classification et régression, de nombreuses méthodes ont été proposées [5] [6]. La méthode DenseWeight propose de pondérer les erreurs pour privilégier les classes sous-représentées. La fonction de perte LMFLoss ajuste les marges en fonction de la distribution des étiquettes.

Présentation du jeu de données

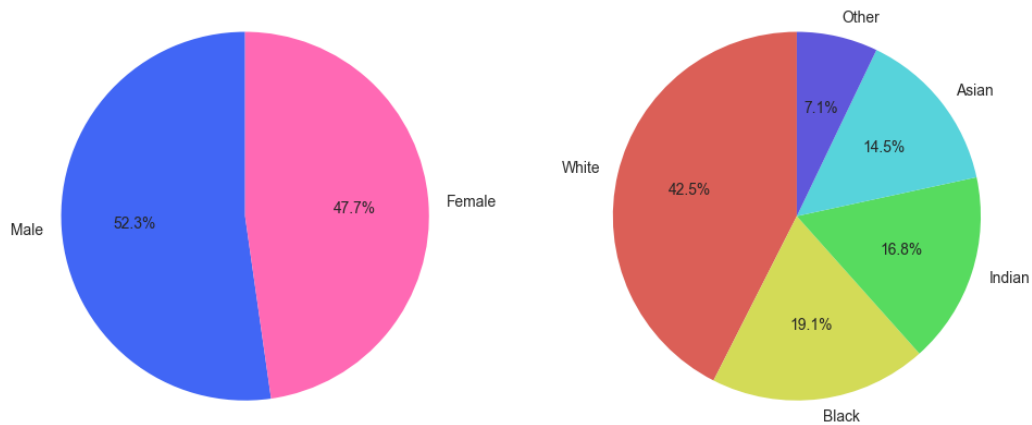
UTKFace est un ensemble de données de visages contenant des images avec des informations d'âge, de genre et d'origine ethnique. Les 23690 images présentes, de dimension de 200x200 pixels et comportant 3 canaux RGB, offrent une grande variété de caractéristiques faciales.

Exemples de visages d'hommes par groupe d'âge



Ce jeu de données présente un bon équilibre entre les genres masculin et féminin. En ce qui concerne les catégories ethniques, la classe « White » (blancs) est surreprésentée, tandis que la classe « Other » (autres) est sous-représentée par rapport aux autres groupes ethniques.

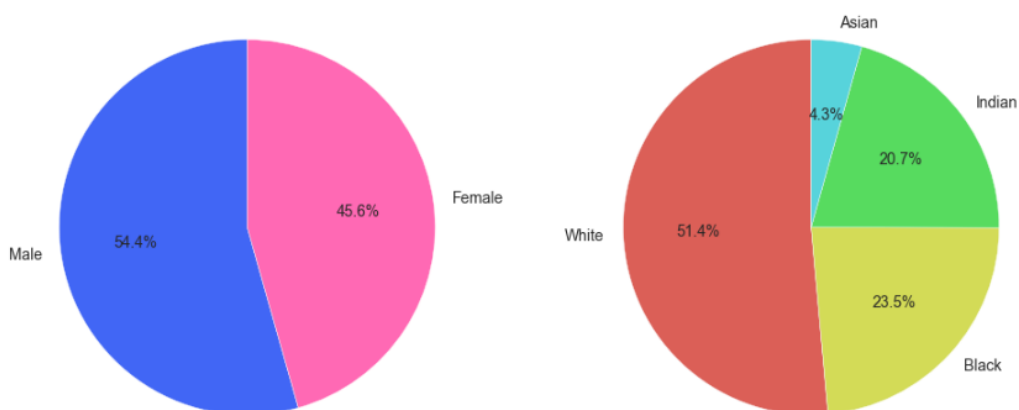
Distribution originale des genres et des origines ethniques



La classe Other rassemble diverses origines ethniques (Hispanique, Latino, Moyen-Orient) qui partagent très peu de similitudes. Les expérimentations menées antérieurement dans ce projet ont révélé que cette catégorie est plutôt difficile à classifier, même en utilisant des méthodes récentes et avancées, en raison de la grande diversité des sous-groupes présents.

Nous décidons de supprimer cette classe du jeu de données et d'effectuer un sous-échantillonnage sur la classe d'origine ethnique « Asian » en utilisant la bibliothèque imbalanced-learn de sklearn. Le sous-échantillonnage est réalisé en supprimant 75% des observations de cette classe, ne gardant ainsi qu'un quart des observations initiales.

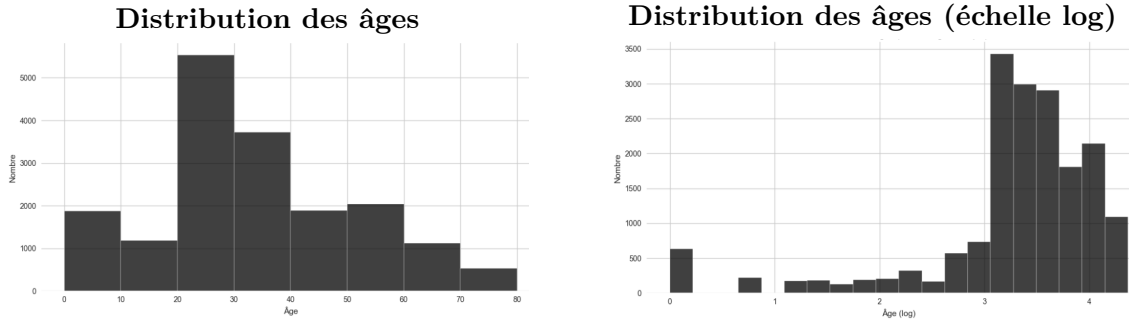
Distribution implémentée des genres et des origines ethniques



Le jeu de données UTKFace présente également un déséquilibre en termes de répartition des âges, avec une concentration importante d'individus âgés de 20 à 40 ans et une représentation limitée des personnes âgées de 60 ans et plus.

Il y a très peu de personnes dont l'âge dépasse 80 ans, en particulier pour les classes 1 et 3. Afin d'assurer la présence de valeurs correspondant à ces âges et ces origines ethniques dans les différents ensembles de données (notamment celui de test qui permettra de comparer les performances), nous décidons de conserver uniquement les photos des personnes dont l'âge est inférieur à 80 ans.

Comparativement aux jeunes, il est plus difficile d'estimer l'âge des personnes âgées ; il est plus facile de distinguer un enfant de 5 ans d'un enfant de 10 ans que de distinguer une personne de 65 ans d'une personne de 70 ans [7]. Ainsi, pour l'estimation d'âge relative d'une personne âgée, il est plus efficace d'utiliser une fenêtre de recherche plus grande. Le processus est modifié en remplaçant chaque âge ϑ par l'âge logarithmique $\log \vartheta$.



En appliquant la transformation logarithmique aux âges, nous avons réussi à obtenir une distribution plus équilibrée des âges chez les adultes, ce qui devrait améliorer les performances de nos modèles en facilitant l'apprentissage et en réduisant les biais.

Préparation des données

La préparation des données est une étape importante pour assurer la performance des modèles de deep learning. Étant donné que MobileNetV3 nécessite des valeurs de pixels entre 0 et 255, nous n'avons pas effectué de prétraitement supplémentaire des images.

Nous avons créé un ensemble de test équilibré en sélectionnant 10 photos par genre, par origine ethnique, par tranches d'âge de 10 ans afin d'obtenir 640 images au total. Nous avons ensuite divisé les 18338 images restantes en deux jeux de données : 80% pour l'entraînement et 20% pour la validation.

Pour l'ensemble d'entraînement, nous avons utilisé un générateur d'images avec des techniques d'augmentation de données, telles que la rotation, le décalage en largeur et en hauteur, la découpe, le zoom et le retournement horizontal, afin de permettre une meilleure généralisation.

Pour les ensembles de validation et de test, nous avons utilisé des générateurs d'images sans augmentation de données. Ils ont été configurés pour lire les chemins d'accès aux images, les attributs à prédire (âge, genre et origine ethnique) et pour générer des lots de 64 images en couleur (RGB) de taille 224x224 pixels.

Cette approche de préparation des données permet de gérer efficacement les ressources de la mémoire lors de l'entraînement et de l'évaluation des modèles, tout en assurant la diversité et la représentativité des données utilisées.

Modèle MobileNetV3

Le modèle MobileNetV3 offre une architecture efficace et éprouvée pour l'extraction de caractéristiques à partir d'images, permettant ainsi de bénéficier d'une phase d'apprentissage plus rapide par rapport à une initialisation aléatoire. Nous avons choisi d'utiliser le modèle minimaliste de MobileNetV3 pour optimiser les performances et réduire la complexité computationnelle.

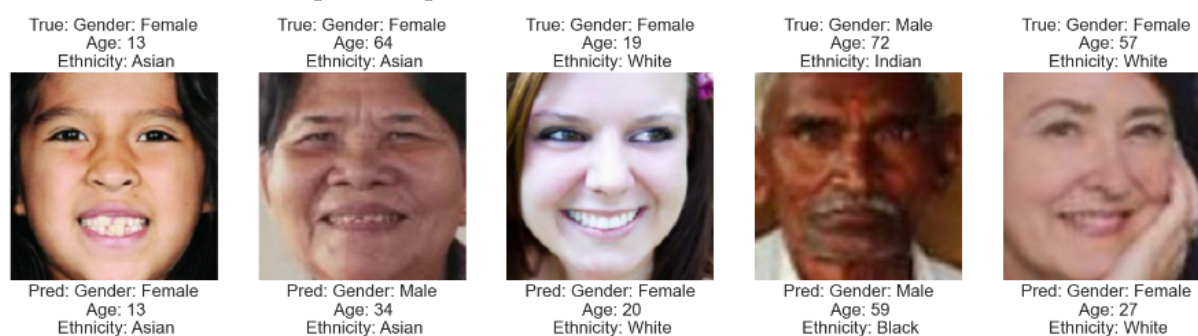
Nous avons déverrouillé toutes les couches pré-entraînées pour les rendre accessibles à l'apprentissage. Ensuite, nous avons ajouté une couche de sortie pour prédire l'âge en utilisant une fonction `< age_output_mod >`, qui effectue une limite des valeurs de sortie dans la plage de 1 à 80. Pour les attributs de genre et d'origine ethnique, nous avons utilisé des couches denses et de régularisation, avec des fonctions d'activation `< sigmoid >` et `< softmax >`, respectivement.

Le modèle a été compilé avec les pertes suivantes : erreur absolue moyenne pour l'âge, entropie croisée binaire pour le genre et entropie croisée catégorielle pour l'origine ethnique. Nous avons attribué des pondérations de perte de 2.0 pour l'âge et l'origine ethnique, et de 1.0 pour le genre. Nous avons choisi avec soin les métriques d'évaluation pour chaque tâche de prédiction. Pour l'âge, qui est une tâche de régression, nous avons utilisé l'erreur absolue moyenne (MAE). Cette métrique quantifie l'erreur moyenne entre les âges prédits et réels, en ignorant la direction de l'erreur. Pour le genre, qui est une tâche de classification binaire, nous avons utilisé la précision (accuracy) comme métrique. Celle-ci représente la proportion de prédictions correctes parmi l'ensemble total des prédictions. Enfin, pour l'origine ethnique, qui est une tâche de classification multiclasse, nous avons utilisé le F1 score macro. Cette métrique calcule la précision et le rappel pour chaque classe puis en fait la moyenne, ce qui donne une mesure équilibrée des performances du modèle.

Une technique d'arrêt précoce a été mise en œuvre pour éviter le surapprentissage et retenir le modèle présentant les meilleures performances.

Après avoir établi ce modèle de base, nous avons utilisé KerasTuner pour rechercher les meilleurs hyperparamètres afin d'en optimiser les performances pendant 25 époques.

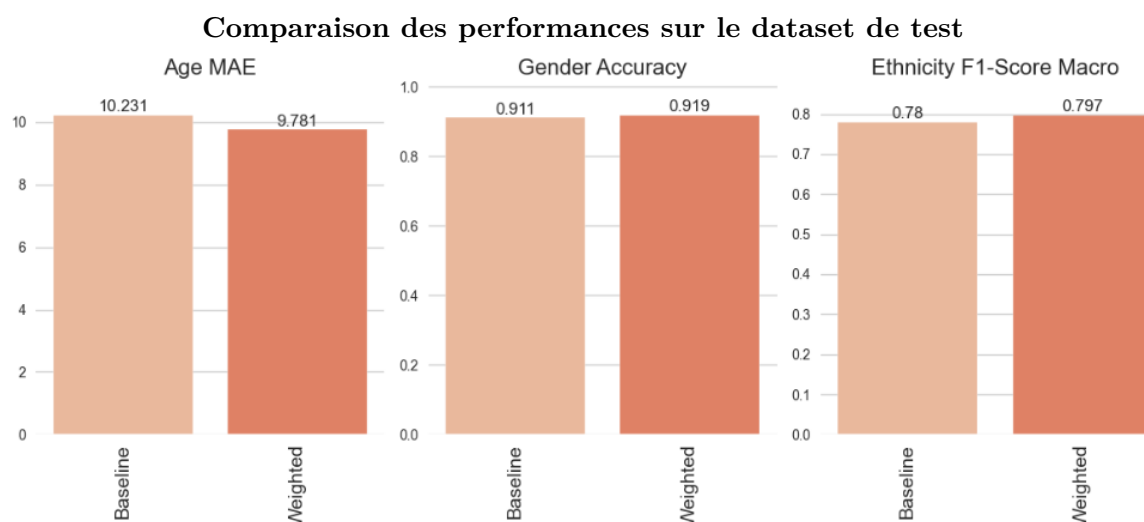
Exemples de prédictions avec le modèle de référence



Nous analyserons par la suite les résultats de ce modèle de référence, en les comparant aux performances d'autres modèles afin de mieux comprendre les forces et les faiblesses de notre approche actuelle.

Méthode usuelle : Pondération des échantillons lors de l'entraînement

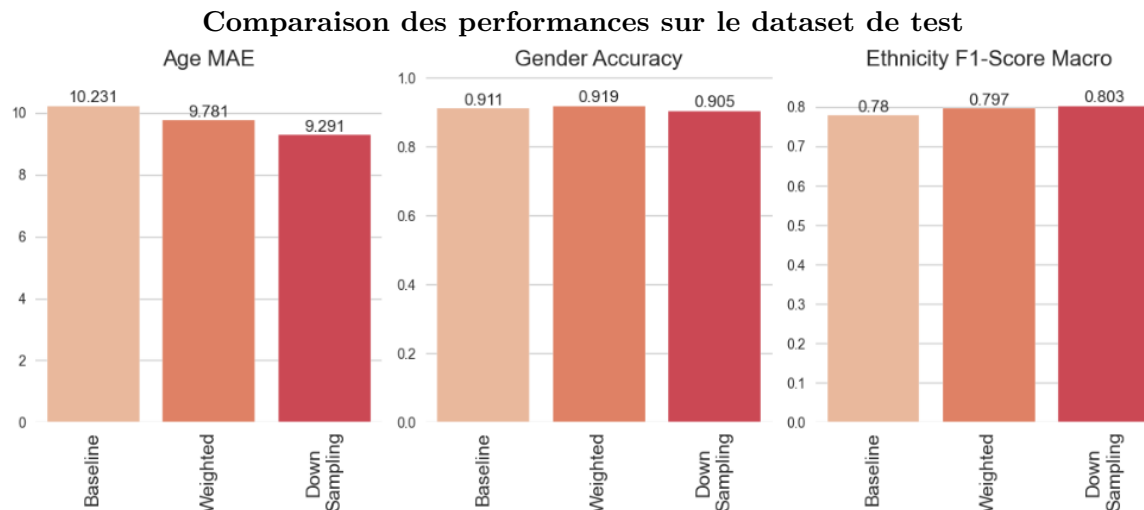
Cette technique est efficace pour gérer les déséquilibres de classe dans les données d'apprentissage. La méthode de l'Inverse du Nombre d'Échantillons (INS) est utilisée pour attribuer un poids plus important aux origines ethniques moins représentées, permettant ainsi au modèle d'apprendre plus efficacement ces classes. Ces poids sont appliqués au générateur de données d'entraînement. Ensuite, le modèle est construit en utilisant les meilleurs hyperparamètres obtenus lors de la précédente recherche KerasTuner et est entraîné avec ces nouvelles pondérations d'échantillons.



Cette méthode a permis une amélioration de 4,4% de la MAE pour l'âge, de 0,9% de l'exactitude du genre et de 2,2% du score F1 pour l'origine ethnique.

Méthode usuelle : Sous-échantillonnage de la classe majoritaire

Une autre approche est essayée pour gérer le déséquilibre des classes : le sous-échantillonnage. Cette technique réduit le nombre d'échantillons de la classe majoritaire pour rééquilibrer la distribution des classes. Ici, la moitié des échantillons de la classe majoritaire (classe 0) sont conservés. Le modèle est ensuite construit en utilisant les meilleurs hyperparamètres obtenus lors de la précédente recherche KerasTuner et est entraîné avec ces données rééchantillonnées.



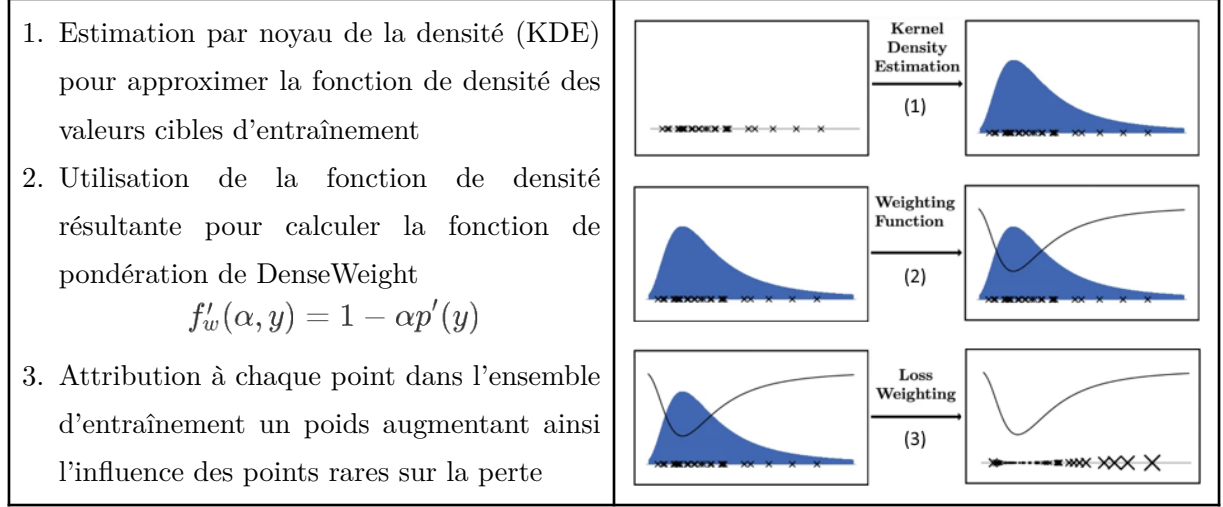
Cette méthode a permis une amélioration de 9,2% de la MAE pour l'âge et de 3,0% du score F1 pour l'origine ethnique. En revanche, l'exactitude du genre a baissé de 0,7%.

DenseWeight

Cette méthode, utilisée dans un contexte de régression, modifie les poids des échantillons lors de l'entraînement et ajuste également la fonction de perte. Elle attribue une importance accrue aux échantillons correspondant à des groupes sous-représentés de valeurs cibles, ce qui oriente l'apprentissage du modèle pour qu'il se focalise davantage sur les erreurs commises sur ces groupes.

Utilisée en conjonction avec la fonction DenseLoss pour calculer l'erreur lors de l'apprentissage, cette technique permet de construire un modèle plus équilibré pour l'ensemble du spectre des âges.

Principe de DenseWeight



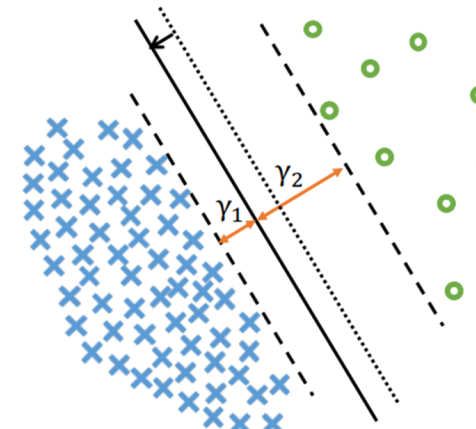
DenseWeight a un hyperparamètre (a) qui permet de configurer dans quelle mesure nous déplaçons l'attention du modèle (poids uniformes pour $a = 0$, plus d'importance sur les échantillons rares quand on augmente a)

Large Margin aware Focal Loss (LMFLoss)

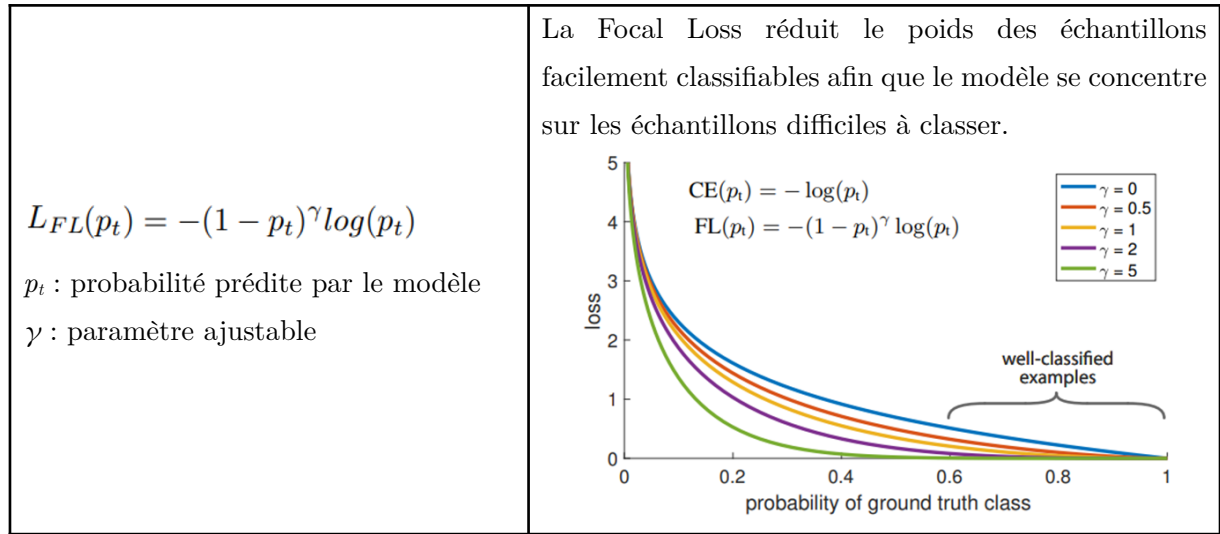
Principe de la LMF Loss

$L_{LMF} = \alpha L_{LDAM} + \beta L_{FL}$	<p>Cette fonction de perte combine linéairement la Focal Loss [8] et la LDAMLoss [9] pour atténuer le déséquilibre de la distribution entre les classes dans les tâches de classification.</p> <p>Deux nouveaux paramètres sont introduits ici, a et β, qui serviront à contrôler l'intensité de l'attention portée aux deux fonctions de pertes.</p>
--	---

Principe de la LDAM Loss

$L_{LDAM}((x, y), f) = -\log \frac{u}{u + \sum_{j \neq y} e^{z_j}}$ $u = e^{z_y - \Delta_y}$ $z_y = f(x)_y$ $\Delta_j = \frac{C}{n_j^{1/4}}$ <p>C : constante n_j : nombre d'observations dans la classe</p>	<p>La LDAM Loss augmente la marge entre les scores de la classe majoritaire et les scores des autres classes.</p>  <p>The diagram shows a scatter plot with blue crosses and green circles. A solid line represents the decision boundary, and a dashed line represents the margin. The distance from the decision boundary to the nearest points of the other class is labeled γ_1 and γ_2.</p>
---	--

Principe de la Focal Loss

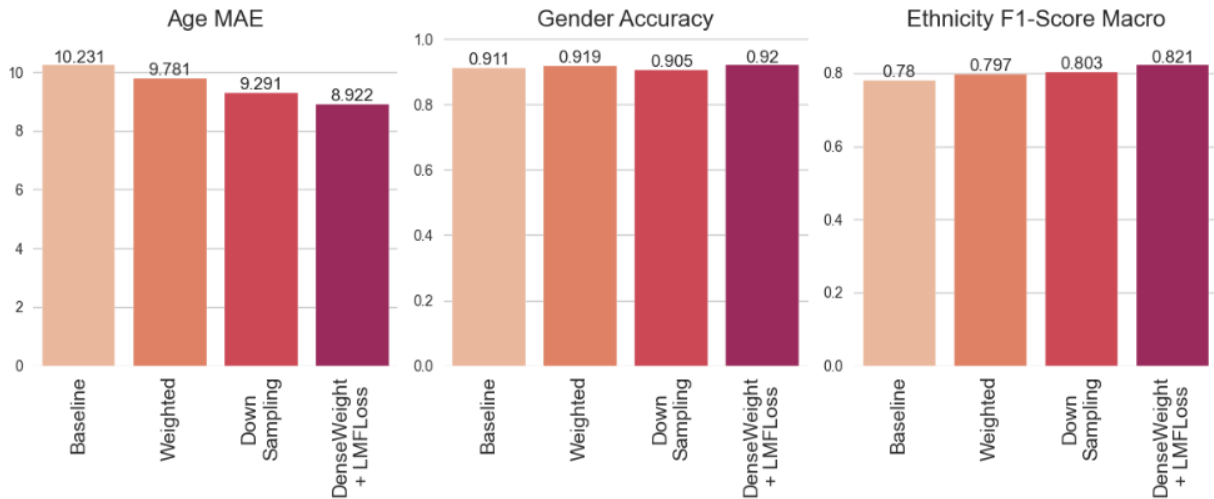


Implémentation de DenseWeight & LMF Loss

Nous avons implémenté ces méthodes dans notre modèle en utilisant le package DenseWeight, en créant une classe DenseLoss et une classe LMF Loss. En utilisant BayesianOptimization, nous avons optimisé les hyperparamètres et obtenu les meilleurs scores pour les valeurs suivantes :

- DenseWeight $a = 0.25$; DenseLoss : $a = 0.25$
- LMF Loss : $a = 2.5$, $\beta = 0.05$ et $\gamma = 4.0$

Comparaison des performances sur le dataset de test



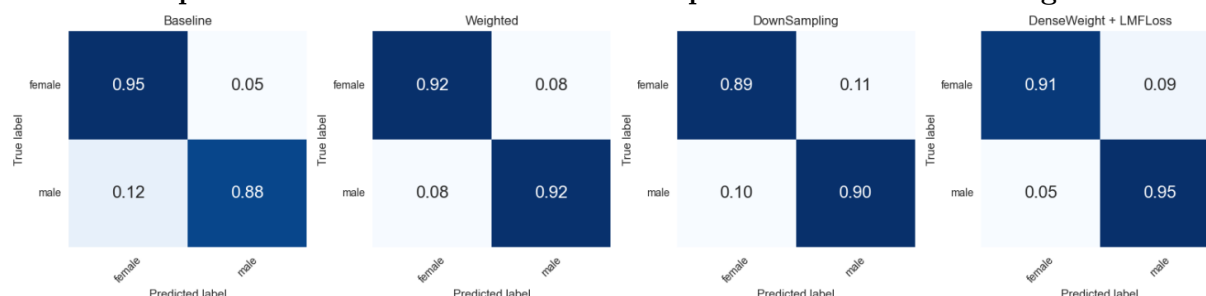
Exemples de prédictions avec le modèle optimisé



Analyse des résultats

- Prédiction du genre

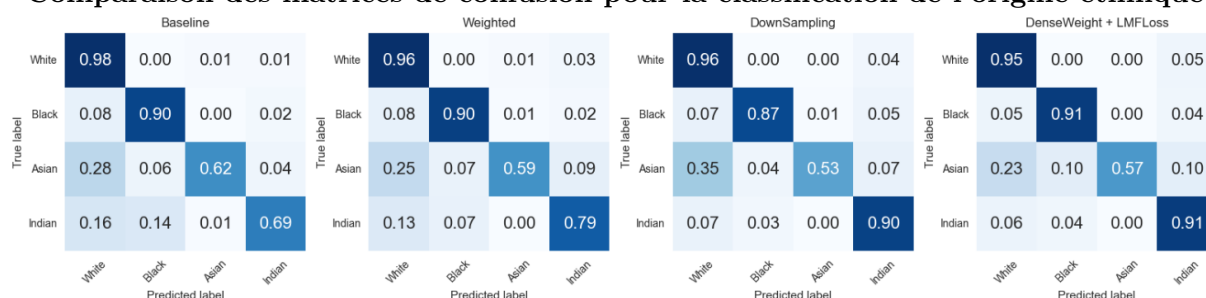
Comparaison des matrices de confusion pour la classification du genre



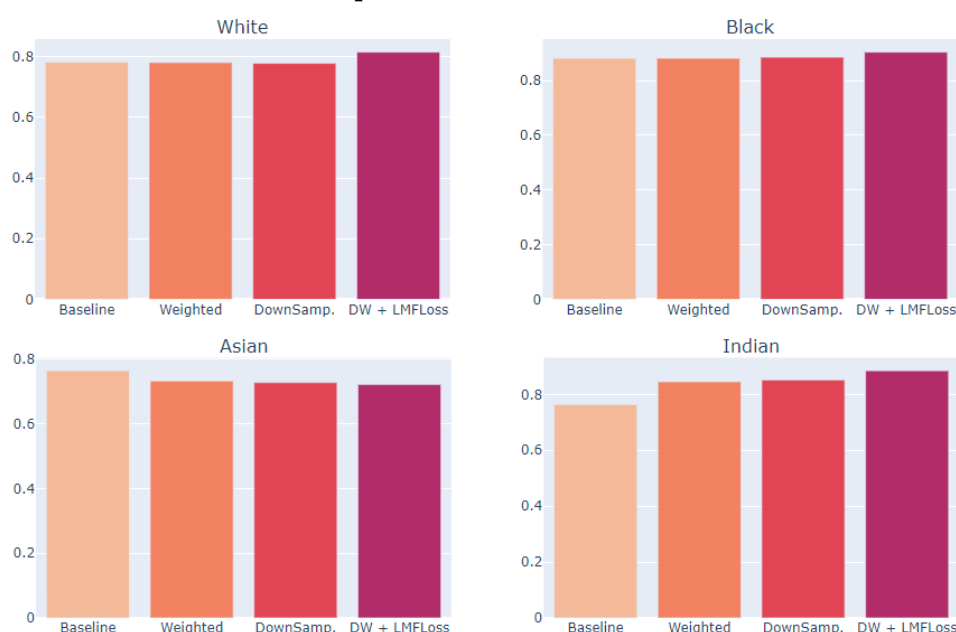
En observant les résultats, il apparaît que les méthodes appliquées n'ont pas eu d'impact significatif sur la classification du genre. En effet, les variations de l'accuracy, qui sont de l'ordre de plus ou moins 1% avec les différentes méthodes, peuvent être considérées comme négligeables. De plus, il est important de noter que dans ce cas particulier, il n'y avait pas de déséquilibre initial concernant cette caractéristique. Par conséquent, il semble que l'application de ces méthodes n'a pas contribué de manière substantielle à améliorer la précision de la classification du genre.

- Prédiction de l'origine ethnique

Comparaison des matrices de confusion pour la classification de l'origine ethnique

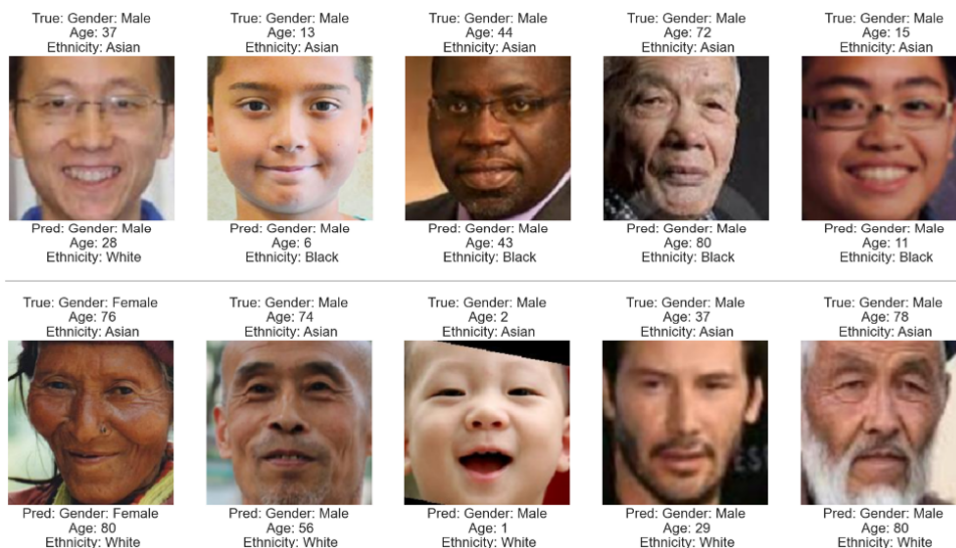


Comparaison des F1 Scores



Concernant la classification de l'origine ethnique, la méthode mise en œuvre n'a pas résolu le problème lié à la prédiction de la classe « Asian », comme en témoigne le maintien d'un score F1 bas. Cependant, elle a permis d'améliorer notablement les scores des trois autres classes. Par conséquent, bien que l'approche employée ait montré des limites dans la prédiction de certaines classes, elle a néanmoins contribué à une meilleure distinction des autres catégories ethniques.

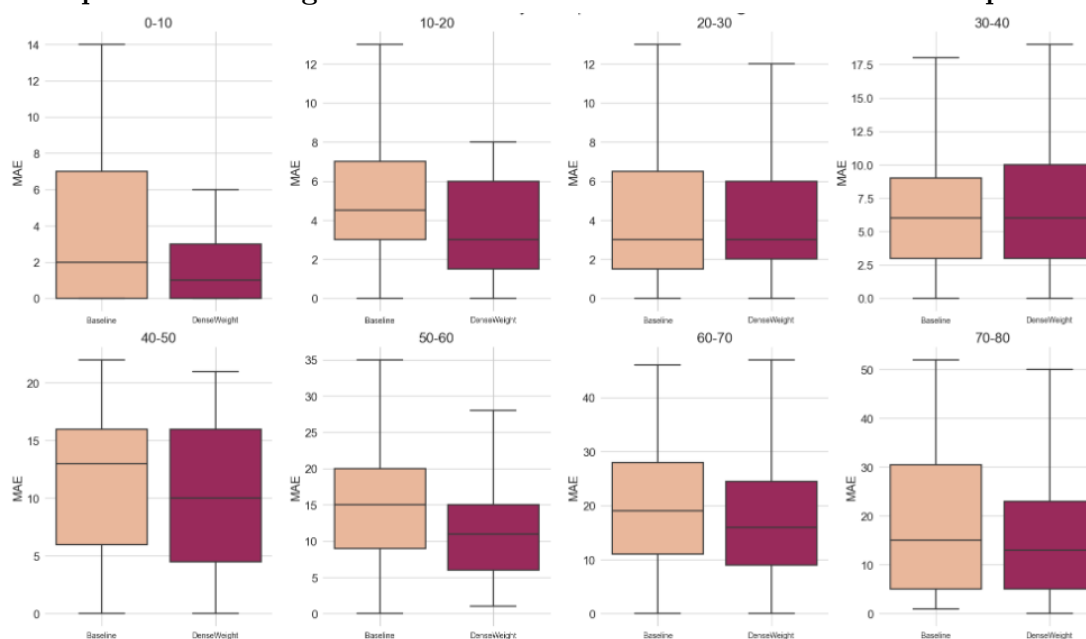
Exemples d'erreurs de classification pour l'origine ethnique : Asian



Nous remarquons qu'il y a des inexactitudes dans la classification de l'origine ethnique. Par exemple, nous pouvons voir sur la première ligne une image qui a été incorrectement étiquetée comme « Asian » alors qu'elle aurait dû être classée comme « Black ». De plus, une autre image sur la seconde ligne montre l'acteur Keanu Reeves, dont l'héritage est plus complexe : bien que sa mère soit d'origine chinoise, son père est américain.

• Prédiction de l'âge

MAE par tranche d'âge entre le modèle de référence et le modèle implémenté



En ce qui concerne l'estimation de l'âge, l'impact de la méthode DenseWeight a été significatif, entraînant une amélioration notable de 12.8%. Cette méthode a particulièrement bien fonctionné pour les tranches d'âges extrêmes. Elle a permis d'améliorer considérablement la précision des prédictions pour les personnes les plus jeunes, dans la tranche d'âge de 0 à 20 ans, mais aussi pour les personnes âgées de plus de 70 ans. Ainsi, cette approche a démontré une efficacité particulière dans la correction des estimations d'âge aux deux extrémités du spectre.

Conclusions

Les résultats indiquent que notre modèle optimisé offre une amélioration significative par rapport à MobileNetV3, en particulier pour l'estimation de l'âge. Cependant, en ce qui concerne la classification des origines ethniques sous-représentées, les performances sont plus nuancées. En effet, LMFloss a eu peu d'impact sur l'amélioration de la classe « Asian », malgré une amélioration notable des performances pour les autres classes.

Il est toutefois important de souligner qu'il reste des opportunités pour améliorer ces performances. L'investigation de techniques plus sophistiquées, telles que les Réseaux Antagonistes Génératifs (GANs), qui sont capables de générer des données supplémentaires pour l'entraînement des classes sous-représentées, pourrait offrir des perspectives intéressantes pour optimiser davantage les performances du modèle.

Bibliographie

- [1] S. Y. Zhang Zhifei, and H. Qi, "Age progression/regression by conditional adversarial autoencoder," 2017. [Online]. Available: <https://susanqq.github.io/UTKFace/>
- [2] A. Howard, M. Sandler, et al., "Searching for mobilenetv3," 2019. [Online]. Available: <https://arxiv.org/pdf/1905.02244.pdf>
- [3] P. D. Michael Steininger Konstantin Kobs, "Density-based weighting for imbalanced regression," 2021. [Online]. Available: <https://link.springer.com/article/10.1007/s10994-021-06023-5>
- [4] A. A. Sadi, L. Chowdhury, et al., "Lmfloss: a hybrid loss for imbalanced medical image classification," 2022. [Online]. Available: <https://arxiv.org/pdf/2212.12741.pdf>
- [5] X. Long, H. Zhang, and T. Zhang, "Label-imbalanced and group-sensitive classification under overparameterization," 2021. [Online]. Available: <https://arxiv.org/abs/2103.01550>
- [6] Y. Yang, K. Zha, Y.-C. Chen, H. Wang, and D. Katabi, "Delving into deep imbalanced regression," 2021. [Online]. Available: <https://arxiv.org/abs/2102.09554>
- [7] N.-H. Shin, S.-H. Lee, and C.-S. Kim, "Moving window regression: a novel approach to ordinal regression," 2022. [Online]. Available: <https://arxiv.org/pdf/2203.13122v1.pdf>
- [8] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," 2018. [Online]. Available: <https://arxiv.org/pdf/1708.02002.pdf>
- [9] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," 2019. [Online]. Available: <https://arxiv.org/abs/1906.07413>