PROJET 8

Participez à une compétition Kaggle !

Avril 2021

Maud Charbonneau

Sommaire

Introduction 3
Compétition 3
Données 4
Analyse exploratoire 4
Anomalies 4
Analyse des classes 5
Analyse des annotations par radiologue 6
Corrélations entre classes 6
Bounding boxes 7
Métadonnées 8
Modélisation 9
Classification et Détection 10
Implémentation des modèles 10
Comparaison avec la baseline 11
Conclusion - Améliorations possibles 12
Ressources 12

Introduction

Le monde de la technologie évolue à une vitesse incroyable. En permanence, de nouvelles technologies et de nouveaux outils apparaissent. Pour cette raison, pour travailler dans ce domaine, il faut avoir la capacité de se tenir à jour. De plus, il existe tellement d'innovations qu'il est difficile pour une personne de savoir tout utiliser. Il faut donc pouvoir d'interagir avec les différentes communautés afin de résoudre un problème, un blocage, trouver des solutions ou développer de nouveaux produits.

Kaggle est un site organisant des compétitions de data science. Des entreprises peuvent proposer des problèmes dans ce domaine et tout le monde peut y participer, individuellement ou en groupe. Les participants obtenant les meilleures performances reçoivent un prix.

Le but de ce projet est donc de participer à une compétition activement afin d'interagir avec le travail d'autres personnes ainsi qu'avec une communauté. Il faut aussi analyser le travail déjà proposé et être capable d'apporter une valeur ajoutée.

Compétition

La compétition que j'ai choisie s'intitule : VinBigData Chest X-ray Abnormalities Detection. Bien que la date limite soit très proche et ne me laissait pas beaucoup de temps, c'est la compétition qui m'a le plus intéressée vu l'enjeu du projet. Elle est proposée par Vingroup Big Data Institute, qui a été fondé en 2018 par Vingroup JSC. Leur but est de promouvoir des recherches fondamentales et étudier de nouvelles technologies développables en se concentrant sur la data science et l'intelligence artificielle.

L'objectif de cette compétition est d'automatiquement localiser et classifier 14 types d'anomalies thoraciques à partir de radiographies de la cage thoracique.

Les radiographies sont énormément utilisées par les médecins pour différents problèmes médicaux. Les radiographies thoraciques font partie des plus difficiles à interpréter. Cela peut entrainer de mauvais diagnostics, même pour un très bon spécialiste. Le but derrière cette compétition est de créer un système de détection et de diagnostics assisté par ordinateur pour permettre de réduire la pression mise sur les médecins vis-à-vis du diagnostic dans les hôpitaux métropolitains bondés en leur apportant une seconde opinion. De plus cela pourrait également aider en zone rurale à améliorer la qualité des diagnostics, ces endroits ayant moins de spécialistes.

Le résultat à fournir pour la compétition est un document CSV contenant un dataframe indiquant l'identifiant de la photo, la classe correspondant à l'anomalie, le score de confiance ainsi que les dimensions de la bounding box soit xmin, ymin, xmax, ymax. Pour une image sans anomalie il faudrait donc retourner : 14 1.0 0 0 1 1.

Données

Les radiographies constituent un dataset de 18 000 scans qui ont été annotés par des radiologistes expérimentés. Le training-set contient 15 000 scans et le test-set contient 3 000 scans au format DICOM.

Le dataframe de train contient huit colonnes : l'identifiant de l'image, le nom de la classe d'anomalie correspondante, le numéro attribué à cette classe, l'identifiant du radiologue ayant annoté le scan, et enfin xmin, ymin, xmax et ymax correspondant aux dimensions de la bounding box contenant l'anomalie.

	image_id	class_name	class_id	rad_id	x_min	y_min	x_max	y_max
0	50a418190bc3fb1ef1633bf9678929b3	No finding	14	R11	NaN	NaN	NaN	NaN
1	21a10246a5ec7af151081d0cd6d65dc9	No finding	14	R7	NaN	NaN	NaN	NaN
2	9a5094b2563a1ef3ff50dc5c7ff71345	Cardiomegaly	3	R10	691.0	1375.0	1653.0	1831.0

Dataframe des données train

Analyse exploratoire

Anomalies

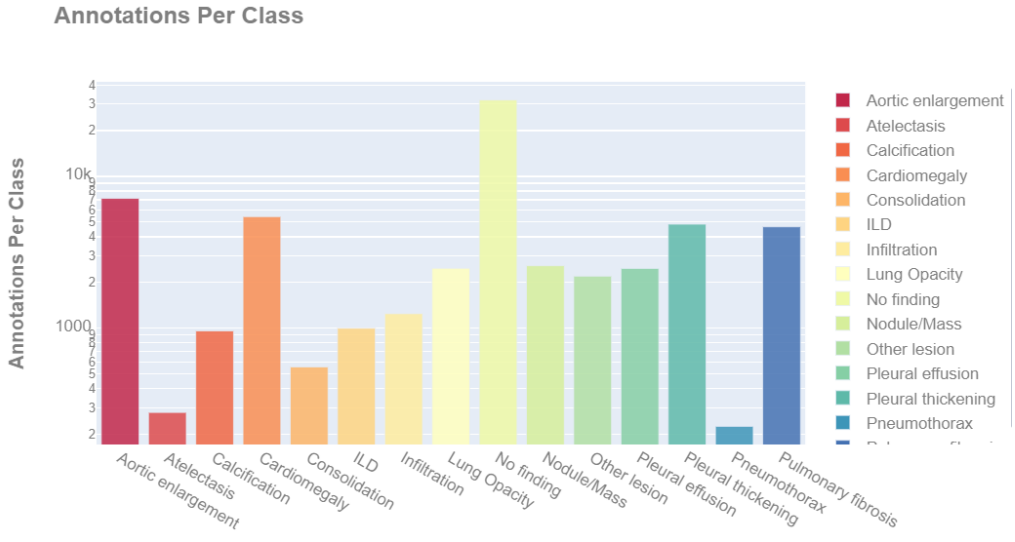
Pour commencer, il faut savoir que les anomalies sont regroupées en 15 classes notées de 0 à 14.

0. **Hypertrophie aortique :** "Un renflement anormal qui se produit dans la paroi du vaisseau sanguin principal."
1. **Atélectasie :** "Effondrement d'une partie du poumon dû à une diminution de la quantité d'air dans les alvéoles entraînant une perte de volume et une augmentation de la densité."
2. **Calcification :** « Dépôt de sels de calcium dans les poumons ». - "[...] des calcifications surviennent dans un poumon endommagé suite à un processus inflammatoire tel qu'une infection (tuberculose, histoplasmose, pneumocystis carinii), un saignement ou un infarctus pulmonaire"
3. **Cardiomégalie :** « L'élargissement du cœur se produit lorsque le cœur d'un patient adulte est plus gros que la normale et que le rapport cardiothoracique est supérieur à 0,5. »

4. **Consolidation :** "Tout processus pathologique qui remplit les alvéoles de liquide, de pus, de sang, de cellules (y compris les cellules tumorales) ou d'autres substances entraînant des opacités lobaires, diffuses ou multifocales mal définies."
5. **Pneumopathie interstitielle :** "Maladie pulmonaire interstitielle ou pneumopathie interstitielle - Implication du tissu de soutien du parenchyme pulmonaire entraînant des opacités réticulaires fines ou grossières ou de petits nodules."
6. **Infiltration :** "Une substance anormale qui s'accumule progressivement dans les cellules ou les tissus corporels ou toute substance ou type de cellule qui se produit à l'intérieur ou se propage à travers les interstices (interstitium et / ou alvéoles) du poumon, qui est étrangère au poumon, ou qui s'accumule en quantité supérieure à la normale. "
7. **Opacité pulmonaire :** "Toute opacité ou opacité focale anormale ou généralisée dans les champs pulmonaires (étiquette de couverture comprenant, mais sans s'y limiter, la consolidation, la cavité, la fibrose, le nodule, la masse, la calcification, l'épaississement interstitiel, etc.)."
8. **Nodule / Masse :** "Tout espace occupant une lésion solitaire ou multiple."
9. **Autres lésions :** "Autres lésions qui ne figurent pas sur la liste des signes ou anomalies mentionnés ci-dessus."
10. **Épanchement pleural :** « Accumulation anormale de liquide dans l'espace pleural ».
11. **Épaississement pleural** : "Toute forme d'épaississement impliquant la plèvre pariétale ou viscérale."
12. **Pneumothorax :** "La présence de gaz (air) dans l'espace pleural."
13. **Fibrose pulmonaire :** « Un excès de tissu conjonctif fibreux dans le poumon ».
14. **Aucune constatation :** explicite, il n'y a pas eu de constatations.

Analyse des classes

Pour commencer, j'ai voulu regarder si certaines catégories apparaissaient plus que d'autres.

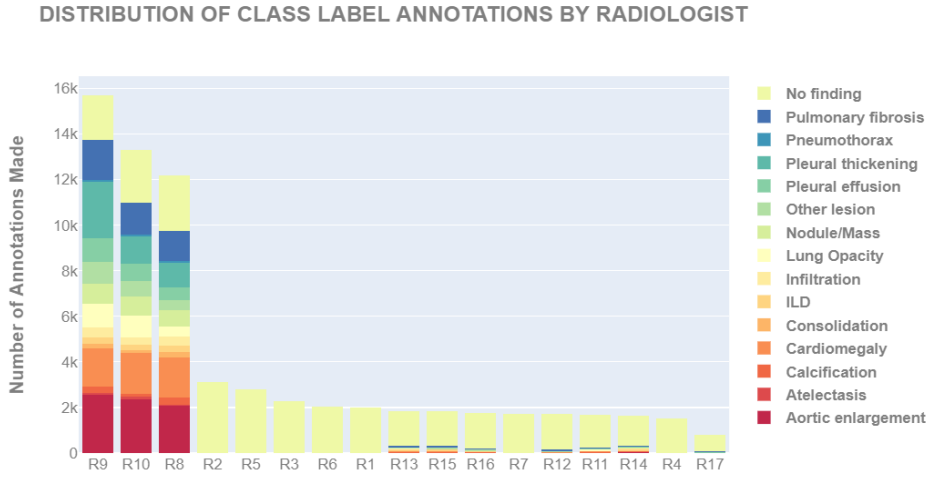


Nombres d'annotations par classe

On remarque que la catégorie 14 est beaucoup plus présente que les autres : 31 818 fois enregistrées contre 7 162 pour la seconde catégorie la plus présente.

Analyse des annotations par radiologue

En tout, 17 radiologues ont annoté les scans. Il serait intéressant de savoir ce qu'ils ont noté, si certains ont trouvé plus d'anomalies que d'autres etc....



Quantification par classes annotés par radiologues

On voit clairement apparaitre sur le graphique ci-dessus que trois praticiens sont responsables de la majorité des annotations des scans (environ 50%). Dans ceux restants, 11 n'ont trouvé aucune anomalie dans tous leurs scans. Les sept spécialistes restants ont détecté différentes anomalies mais la majorité de leurs annotations est également "Aucune constatation". Cela démontre l'utilité d'un système de détection et de diagnostic assisté par ordinateur afin d'éviter les diagnostics erronés.

Corrélations entre classes

La prochaine étape est de vérifier s'il y a des corrélations entre les différentes anomalies.

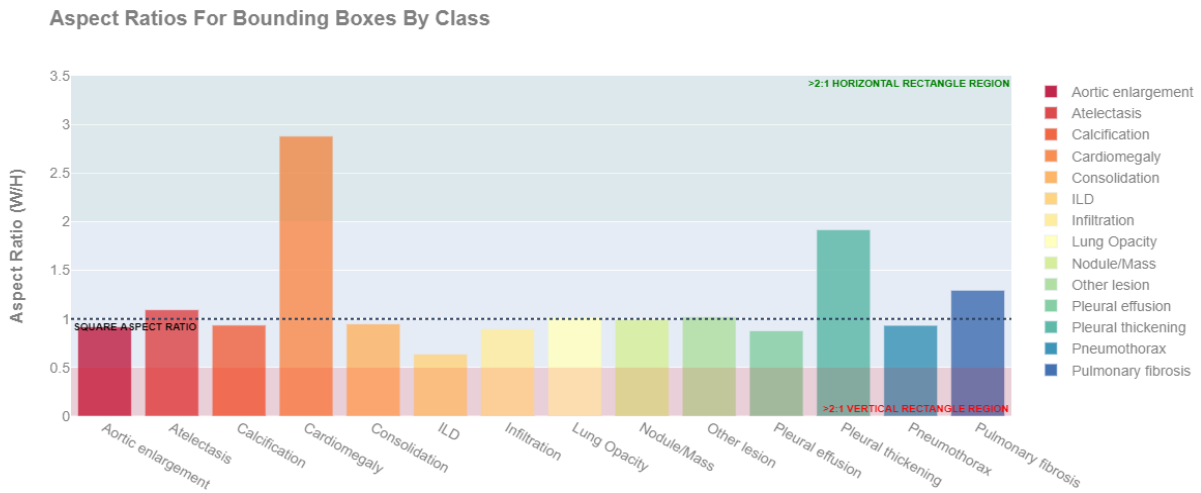
class_id	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	1.00	0.05	0.20	0.69	0.09	0.17	0.15	0.29	0.24	0.31	0.24	0.46	0.03	0.32	-0.79
1	0.05	1.00	0.10	0.02	0.21	0.07	0.13	0.18	0.09	0.16	0.18	0.18	0.11	0.21	-0.17
2	0.20	0.10	1.00	0.11	0.04	0.10	0.03	0.10	0.28	0.21	0.09	0.22	0.00	0.23	-0.27
3	0.69	0.02	0.11	1.00	0.06	0.07	0.08	0.20	0.12	0.24	0.14	0.30	0.01	0.17	-0.66
4	0.09	0.21	0.04	0.06	1.00	0.06	0.28	0.41	0.20	0.14	0.29	0.19	0.13	0.21	-0.24
5	0.17	0.07	0.10	0.07	0.06	1.00	0.26	0.14	0.10	0.15	0.13	0.20	0.01	0.23	-0.25
6	0.15	0.13	0.03	0.08	0.28	0.26	1.00	0.38	0.11	0.16	0.24	0.24	0.03	0.38	-0.32
7	0.29	0.18	0.10	0.20	0.41	0.14	0.38	1.00	0.33	0.34	0.42	0.37	0.13	0.40	-0.48
8	0.24	0.09	0.28	0.12	0.20	0.10	0.11	0.33	1.00	0.29	0.17	0.26	0.04	0.27	-0.38
9	0.31	0.16	0.21	0.24	0.14	0.15	0.16	0.34	0.29	1.00	0.29	0.35	0.13	0.32	-0.44
10	0.24	0.18	0.09	0.14	0.29	0.13	0.24	0.42	0.17	0.29	1.00	0.55	0.18	0.33	-0.42
11	0.46	0.18	0.22	0.30	0.19	0.20	0.24	0.37	0.26	0.35	0.55	1.00	0.12	0.48	-0.61
12	0.03	0.11	0.00	0.01	0.13	0.01	0.03	0.13	0.04	0.13	0.18	0.12	1.00	0.06	-0.12
13	0.32	0.21	0.23	0.17	0.21	0.23	0.38	0.40	0.27	0.32	0.33	0.48	0.06	1.00	-0.54
14	-0.79	-0.17	-0.27	-0.66	-0.24	-0.25	-0.32	-0.48	-0.38	-0.44	-0.42	-0.61	-0.12	-0.54	1.00

Table de corrélation entre les classes

La classe “Aucune constatation” est négativement corrélée au reste ce qui est logique. Les autres anomalies sont légèrement corrélées entre elles, à l’exception de l’hypertrophie aortique et de la cardiomégalie qui sont fortement corrélées.

Bounding boxes

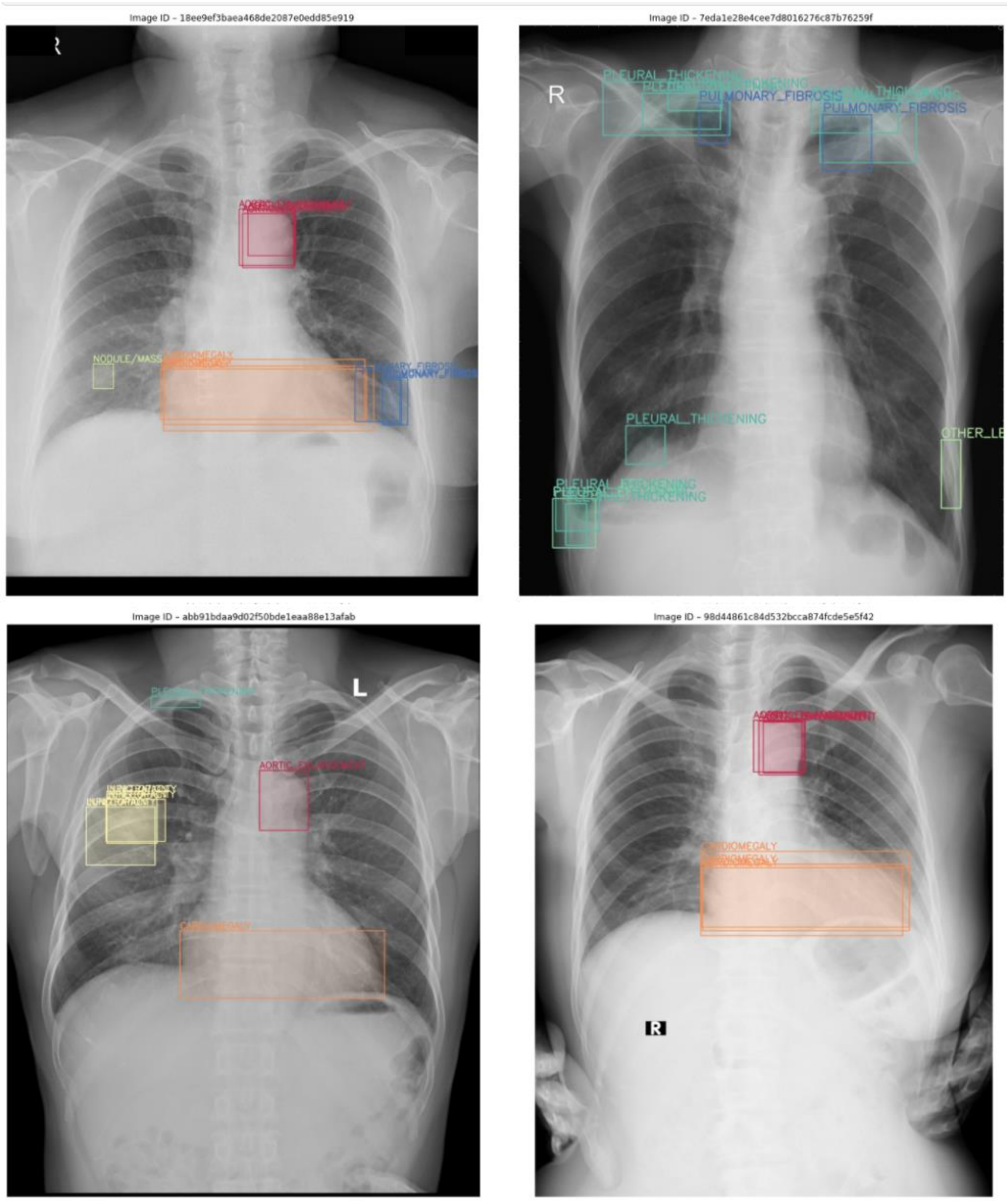
On souhaite maintenant essayer de définir certaines formes des boundings boxes associées à chaque classe. Les boundings boxes sont des cadres qui délimitent certains objets choisis sur les images. Ici, elles permettent de détecter les anomalies. Pour tenter de définir leurs formes on utilise un graphique.



Aspect des boundings boxes par classe

Les bounding boxes qui se distinguent le plus sont la Cardiomégalie qui a une boite en moyenne rectangulaire fine et horizontale et la Pneumopathie interstitielle (ILD) qui a une boite rectangulaire fine et verticale (hauteur 1,6x plus grande que largeur).

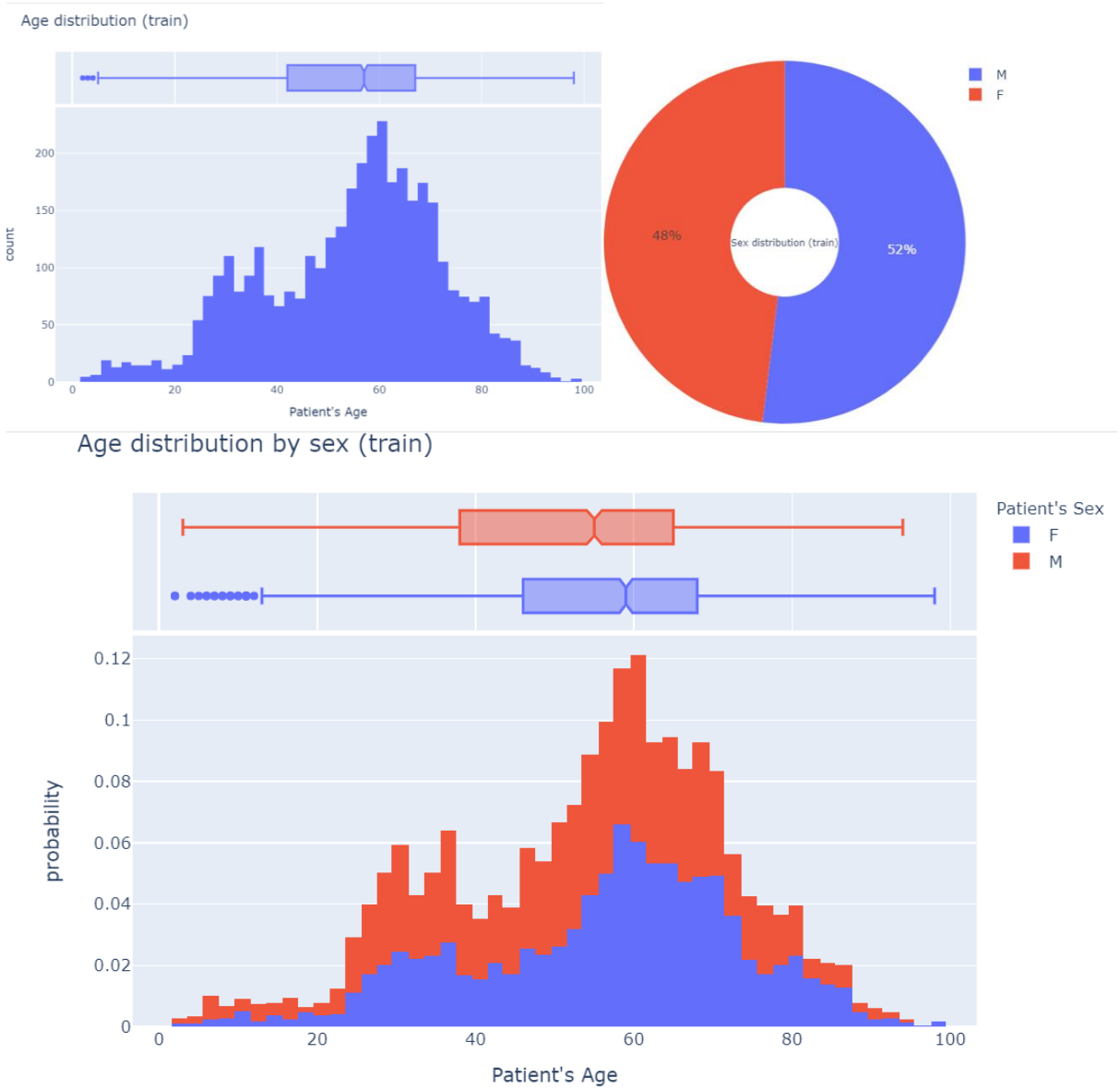
On utilise les valeurs contenues dans la base train pour dessiner les boundings boxes des anomalies sur différentes images. On remarque que certaines classes possèdent des cadres très différents entre eux et situés à différentes parties de la cage thoracique. D'autres classes sont situées chaque fois à des endroits très proches et ont des proportions similaires.



Bounding boxes par classe affichées sur les radiographies

Métadonnées

Pour finir l'EDA et avant de passer à la modélisation, il y a une autre base de données potentiellement utile à l'étude : les métadonnées. Elles regroupent quatre variables : l'âge du patient, son sexe, sa taille et son poids. Ces valeurs ne sont pas associées à une classe en particulier mais à toutes les anomalies.



Distribution des âges, des sexes et des âges par sexe

Ces données permettent d'observer que, de manière générale, le pic d'anomalie est aux alentours des 60 ans. Il est, cependant, presque tout au long de la vie des patients, deux fois plus important pour les hommes. Aux alentours des 30 ans, la probabilité d'avoir une anomalie augmente fortement, avant de redescendre lors de la quarantaine. Elle remonte de nouveau vers les 50 ans afin d'atteindre son maximum à 60 ans puis de diminuer progressivement.

Modélisation

Classification et Détection

Le problème à résoudre dans cette compétition est de localiser et classifier en 14 classes les anomalies présentes sur les radiographies. La manière dont est évaluée la compétition est avec le mean Average Precision (mAP). Cette métrique permet de calculer le score obtenu. En lisant différents notebooks et certaines discussions de la compétition, j'ai appris que ce score s'améliorait en utilisant une classification à deux classes : anomalie ou aucune anomalie. Pour augmenter le score du projet il faudrait donc faire premièrement une classification à deux classes, puis une détection avec un R-CNN afin de localiser et classifier les anomalies. Les R-CNN (Region Based Convolutional Neural Networks traduit par Réseaux de Neurones Convolutifs basés sur les Régions) sont une famille de modèles d'apprentissage automatique pour la vision par ordinateur et en particulier la détection d'objets. Ce type de famille est nouveau pour moi, je ne l'ai pas encore étudié. Pour cette raison, bien que cette partie soit importante pour le projet, j'ai décidé de concentrer ce projet sur l'EDA et la classification à deux classes uniquement et ne pas réaliser la détection. Cependant j'ai regardé de nombreux notebooks sur le sujet et l'un des outils souvent employé est Yolo. Yolo, qui veut dire "You Only Look Once", est un réseau de neurones spécialisé dans la détection et l'analyse d'objets dans l'image. Sa grande force est la rapidité : il peut travailler en temps réel (à 45 im / sec). Il est plus rapide que des R-CNN, car il découpe l'image en petits blocs et génère des tenseurs pour chaque bloc.

Implémentation des modèles

J'ai également trouvé un notebook effectuant cette démarche utilisant un modèle de EfficientNet. J'ai déjà travaillé dans un précédent projet avec ce groupe de CNN mais l'implémentation était rapide puisque j'avais utilisé Keras qui rend cette partie plus simple. Mais dans ce notebook, c'est PyTorch qui était utilisé donc j'ai décidé de le reprendre pour me familiariser avec cette nouvelle bibliothèque.
Le modèle original utilisé était EfficientNet-B3 - NoisyStudent. NoisyStudent Training est une méthode d'apprentissage semi-supervisée obtenant de bons résultats sur ImageNet et permet d'augmenter les performances de EfficientNet sur ces données. La base de données ne contenant que 18 000 données, j'ai décidé d'utiliser un modèle de EfficientNet plus petit : c'est pour cela que j'ai utilisé EfficientNet-B0.

Ce qu'on souhaite obtenir est un modèle qui définisse si l'image d'entrée appartient à la classe "anomalie" ou "pas d'anomalie" ainsi que la probabilité d'appartenance à cette classe.

	image_id	class0	class1
0	8dec5497ecc246766acfba5a4be4e619	0.976988	0.023012
1	287422bed1d9d153387361889619abed	0.950402	0.049598
2	1d12b94b7acbeadef7d7700b50aa90d4	0.995952	0.004048
3	6b872791e23742f6c33a08fc24f77365	0.874948	0.125052
4	d0d2addff91ad7beb1d92126ff74d621	0.997519	0.002481

Tableau de probabilité de présence ou non d'anomalie

Le modèle que j'ai choisi est donc EfficientNet-B0. Pour importer le modèle, j'ai utilisé un module nommé Timm. PyTorch Image Models (ou timm) est une collection de modèles d'images, de couches, d'utilitaires, d'optimiseurs, et autres, visant à rassembler une grande variété de modèles SOTA. Pour comparer la performance de mon modèle, je le compare à celle du notebook original soit EfficientNet-B3 NoisyStudent.

Comparaison avec la baseline

Voici donc les performances de ce modèle et de EfficientNet-B0, pour les mêmes paramètres, sur les données train et test.

	Performances	
	Données train	Données test
EfficientNet-B3 NoisyStudent	0,9810	0,9231
EfficientNet-B0	0,9828	0,8677

Performance du modèle de baseline et du nouveau modèle

Les performances sur les données train sont meilleures pour le nouveau modèle que pour le modèle servant de baseline mais ce n'est pas le cas sur les données test. J'ai donc essayé d'améliorer le résultat avec du Transfert Learning. J'ai testé deux stratégies : le fine-tuning partiel et l'extraction de features. Utilisant PyTorch, le code pour modifier le modèle est très différent qu'avec Keras.

(a)

```
for param in model.parameters():
    param.requires_grad_(False)
ct = 0
for child in model.children():
    ct += 1
    if ct < 8:
        for param in child.parameters():
            param.requires_grad = True
```

(b)

```
# Set up the last 3 layers to not trainable
for layer in model_IV3.layers[:3]:
    layer.trainable = False
```

(c)

```
# Set up the layers to not trainable
for layer in model_IV3.layers:
    layer.trainable = False
```

Comparaison du code de Transfert Learning avec PyTorch et Keras

Le modèle (a) est avec PyTorch. Les deux premières lignes correspondent à l'extraction de features et l'intégralité est le fine-tuning partiel. Pour le fine-tuning partiel, il faut geler toutes les couches puis dégeler celles voulues ensuite. Le modèle (b) correspond au fine-tuning partiel avec Keras et le modèle (c) l'extraction de features.

Voici les résultats obtenus :

	Performances	
	Données train	Données test
Fine-tuning partiel	0,9365	0,9053
Extraction de features	0,9963	0,9603

Performance des différentes stratégies de Transfert Learning

Le nouveau modèle avec extraction de features performe donc mieux que le modèle de baseline: 0,9231 pour le modèle baseline contre 0,9603 pour le nouveau modèle implémenté sur les données test.

Conclusion - Améliorations possibles

- Ne connaissant pas encore suffisamment EfficientNet, il pourrait être intéressant de tester différents modèles entre B0 et B3 par exemple pour savoir lequel performe le mieux avec cette quantité de données.

- Comme dit précédemment, mon projet ne contient qu'une classification. Pour attendre le résultat final attendu par la compétition, il faudrait donc ajouter une détection et utiliser un R-CNN pour cela.

- Ce projet est très différent des autres. Il m'a fallu d'abord me renseigner sur la compétition que je souhaitais faire puis regarder ce qui avait été fait et réfléchir à ce que je pouvais apporter. Dans ce type de compétition, certains des compétiteurs ont un très haut niveau de connaissances, certains travaillent en groupe etc... Réussir à comprendre le travail effectué par d'autres et trouver ce que je pouvais apporter avec mes connaissances s'est avéré assez difficile. Mais cela m'a appris à lire de nombreux notebooks faits par d'autres personnes et m'a également prouvé l'importance des commentaires et des markdowns. Lire un code écrit par une autre personne est beaucoup plus difficile mais les explications en commentaires ou textes permettent de beaucoup mieux comprendre et aussi d'élargir ses compétences.

- J'ai également commencé à me familiariser avec PyTorch ce qui était quelque chose de nouveau pour moi. Le projet précédent m'a aidée à mieux faire mes recherches et à prendre plus rapidement en main les nouveaux outils comme celui-là. J'ai mis à profit l'apprentissage précédent car je n'avais que 14 jours pour participer à la compétition. Le fait de travailler sur une plateforme ayant pour but d'échanger m'a également aidé à avancer plus rapidement sur le projet.

- Je n'ai malheureusement pas eu le temps d'explorer la seconde partie de la modélisation de ce projet. Toutefois j'ai pu me renseigner sur la détection et me familiariser avec différentes manières de réaliser cette étape. Ce projet m'a donc permis de découvrir d'autres parties du Machine Learning que je souhaite apprendre par la suite.

Ressources

<u>Compétition :</u>

- https://www.kaggle.com/c/vinbigdata-chest-xray-abnormalities-detection/

Ma participation :

- https://www.kaggle.com/maudcharbonneau/vinbigdata-chest-x-ray-eda-efficientnetb0

Github :

- https://github.com/maudch96/Projet8

Notebooks utilisés :

- Modèle baseline :
  - https://www.kaggle.com/mrinath/2-class-classifier-pipeline-using-effnet
- EDA :
  - https://www.kaggle.com/dschettler8845/visual-in-depth-eda-vinbigdata-competition-data
  - https://www.kaggle.com/bjoernholzhauer/eda-dicom-reading-vinbigdata-chest-x-ray
  - https://www.kaggle.com/bryanb/vinbigdata-chest-x-ray-eda-fusing-boxes

Autres ressources :

- https://arxiv.org/abs/1911.04252
- https://pypi.org/project/timm/
- https://discuss.pytorch.org/t/how-the-pytorch-freeze-network-in-some-layers-only-the-rest-of-the-training/7088/3
- https://discuss.pytorch.org/t/partial-transfer-learning-efficientnet/109689/3
- https://www.kaggle.com/c/vinbigdata-chest-xray-abnormalities-detection/discussion/208837
- https://www.kaggle.com/awsaf49/vinbigdata-2-class-filter