RAPPORT D’ANALYSE DES ALGORITHMES DE DEEP POUR DISTINGUER DES CAS DE SIGNAUX ECG

**1ère étape : Collecte et analyse des données**

Les donnes sont collectées sous la forme de fichiers tsv, similaires à des csv.

Chaque ligne du fichier correspond à un cas : la première colonne correspond à la classe et le reste sont les données. La classe est soit -1, soit 1.

On se rend compte en analysant les données **que les classes sont déséquilibrées** : il y a plus de cas normaux que de cas anormaux, ce qui est naturel pour ce type de données.

Il y a quasiment **2 fois plus de cas normaux que d’anormaux**. On peut alors déjà suspecter tous les modèles de plus prédire la classe des cas normaux car celle-ci est surreprésentée.

Pour pallier ce problème, il aurait été possible d’effectuer de l’augmentation de données, soit pour augmenter globalement tout le jeu de données et ainsi avoir un échantillon de données plus grand, soit juste augmenter les cas anormaux.

Mais finalement, aucune augmentation de données n’a été faite car nous n’avons pas trouvé des manières « simples » d’augmenter des données ECG. Dans le cas de données images, l’augmentation des données est plus naturelle, mais cela n’est pas le cas pour des ECG.

Les données sont transformées en un X et un y : x étant les données des signaux ECG, y étant la classe. On encode les classes y au format one hot et on normalise les données grâce à l’algorithme minmax, afin de ne pas avoir de valeurs trop extrêmes. De cette manière, le modèle devrait mieux généraliser.  
(En réalité, pour ce jeu de données, on obtient des performances similaires avec et sans normalisation, car les données sont assez « simples ». Mais normaliser les données reste une pratique indispensable.)

**2ème étape : Choix des modèles**

Plusieurs modèles sont entraînés. Les différents modèles sont des MLP, des CNN ou des RNN.

On a d’abord commencé par des modèles plus « simples », avec moins de couches et de neurones. Par exemple pour le 1er MLP, il n’y a qu’une couche cachée de 32 neurones tandis que pour le 3ème MLP, il y a 6 couches cachées de 256 neurones chacune. En **augmentant la complexité du modèle, on souhaite apprendre des fonctions plus complexes, et ainsi reconnaître des données plus précises et mieux généraliser en général.**

La même méthode a été utilisée pour les CNN et les RNN, on entraîne d’abord des réseaux avec peu de couches et de neurones puis on complexifie les modèles après.

De plus, on ajoute des couches de régularisation, de Dropout plus précisément. Ces couches aident les modèles à généraliser en leur compliquant la tâche, et aident aussi à éviter le surentrainement des modèles.

La piste de **l’apprentissage par transfert** a été abordée, mais abandonnée car cela semble trop compliqué pour des données ECG et pour les séries temporelles en général. Cela fait en effet l’objet de papiers (dont un papier publié par Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar and Pierre-Alain Muller) ce qui prouve en quelque sorte la complexité de la tâche.

La piste de l’apprentissage par transfert a été abordée car le jeu de données est assez faible en termes de quantité, et il peut être difficile d’apprendre « from scratch ». Il n’est pas non plus très courant de d’entraîner des modèles from scratch, l’apprentissage par transfert est grandement utilisé pour entraîner des modèles dans la mesure où cela facilite grandement la tâche.  
Le Transfer Learning se fait assez naturellement pour des données images dans la mesure où beaucoup d’architectures de CNN existent avec des poids pré-entraînés sur ImageNet, telles que les modèles VGG (VGG16 ou VGG19), les modèles ResNet (ResNet-34, 50, 101, etc…) ou MobileNetV2. Ces modèles sont facilement adaptables à des cas de traitement d’image (2D) , mais dans le cas de séries temporelles (1D), nous n’avons pas su utiliser de l’apprentissage par transfert car les modèles précédents fonctionnent avec qu’avec des images en 2D.

**3ème étape : Entraînement, résultats et conclusion**

Les modèles ensuite sont évalués à l’aide des métriques **« accuracy » et « f1-score »**, mais nous privilégions la métrique « F1-Score » car le jeu de données est déséquilibré. Le score « accuracy » n’est pas pertinent dans ce cas car nous pourrions avoir un score d’accurayc élevé sans pour autant avoir un modèle qui généralise (on peut prendre comme exemple la détection de fraude dans le milieu bancaire, si on prédit 100 cas normaux alors qu’il n’y en a que 99 et 1 cas de fraude, on atteint une accuracy de 99% mais le modèle n’est pas utile).

Enfin, en plus des métriques d’évaluation, on **évalue aussi le temps d’entraînement** des modèles, ce qui implique aussi leur consommation énergétique.

Les modèles plus complexes prennent plus de temps pour être entraînés et consomment évidemment plus, mais ne sont pas forcément « meilleurs ». Finalement, tous nos modèles atteignent une performance assez similaire aux alentours de 80% de justesse et de score-F1 tandis que les temps d’entraînements sont différents.   
Il s’agira alors de faire un compromis entre la performance du modèle et son temps d’entraînement, choisir le modèle qui a une bonne performance par rapport à son temps d’entraînement.

De plus, dans le cas d’analyse d’ECG**, il faut regarder au-delà des chiffres de métrique**.   
Il est par exemple moins grave de prédire un cas normal comme étant anormal plutôt que l'inverse.

Dans ce cas, ces modèles pourraient pousser le docteur à effectuer plus d'analyses sur le patient pour se rendre compte que le patient n'a rien, alors que dans le cas inverse, cela pourrait entraîner des répercussions grave sur le patient si le docteur ne pousse pas plus les analyses et se repose sur la décision du modèle.

On pourrait alors considérer un modèle comme étant meilleur parce qu’il catégorise bien tous les cas anormaux comme étant anormaux, même si ses métriques de performances ne sont pas les meilleures.

Dans des cas réels d’utilisation de modèles d’apprentissages, les décisions des modèles ne sont jamais à prendre comme étant des vérités absolues.   
Ces outils restent **des outils d’aide à la décision** pour un expert métier, qui pourra s’appuyer sur les prédictions du modèle pour confirmer ou non ses doutes.