RER – Deep Learning

# Contexte :

Se familiariser avec les différents concepts essentiels associés aux Réseaux Neuronaux et Deep Learning, dans un contexte d’apprentissage supervisé et comparer les performances d’algorithmes de Machine Learning (modèles non connexionnistes) et de Deep Learning.

# Problématique(s) :

* Comment construire un réseau de neurones avec TensorFlow et Keras ?
* Comment utiliser ces bibliothèques ?

# Mots clés :

* Neurone artificiel :Un neurone artificiel est une unité de traitement de l'information inspirée des neurones biologiques. Il est généralement utilisé comme élément de base pour la construction de réseaux de neurones artificiels. Un neurone artificiel reçoit des entrées sous forme de signaux numériques, pondère ces entrées en fonction de leurs importances respectives, puis combine les entrées pondérées pour produire une sortie. La pondération des entrées et la fonction de combinaison peuvent être ajustées au moyen d'un processus d'apprentissage supervisé ou non supervisé, en fonction de l'objectif de la tâche à accomplir.
* Réseau de neurones : Un réseau de neurones est un système de traitement de l'information qui s'inspire du fonctionnement du cerveau humain. Il est composé de plusieurs neurones artificiels interconnectés qui travaillent ensemble pour traiter les entrées et produire une sortie. Les connexions entre les neurones sont pondérées, ce qui permet au réseau de modéliser des relations complexes entre les entrées et les sorties. Les réseaux de neurones sont souvent utilisés pour des tâches de classification, de reconnaissance de formes, de prédiction ou de traitement du langage naturel.
* Perceptron :Le perceptron est un type de réseau de neurones artificiels à une seule couche. Il est composé d'un ensemble de neurones artificiels connectés aux entrées et à une sortie. Chaque entrée est pondérée et sommée, puis passe à travers une fonction d'activation pour produire une sortie. Le perceptron est souvent utilisé pour des tâches de classification binaire, c'est-à-dire pour séparer des données en deux catégories distinctes. Il est capable d'apprendre à partir d'exemples grâce à un algorithme d'apprentissage supervisé. Le perceptron est considéré comme le précurseur des réseaux de neurones multicouches plus complexes.
* Multilayer Perceptron (MLP) : Le Multilayer Perceptron (MLP) est un type de réseau de neurones artificiels à plusieurs couches. Il est composé d'une couche d'entrée, une ou plusieurs couches cachées, et une couche de sortie. Chaque couche est composée d'un ensemble de neurones artificiels interconnectés. Les entrées sont propagées de la couche d'entrée à travers les couches cachées, où elles sont pondérées et sommées, puis passées à travers une fonction d'activation. La sortie de chaque couche cachée sert d'entrée à la couche suivante, jusqu'à atteindre la couche de sortie. Le MLP est souvent utilisé pour des tâches de classification, de prédiction ou de reconnaissance de formes. Il peut apprendre à partir d'exemples grâce à un algorithme d'apprentissage supervisé.
* Back Propagation (rétropropagation): La rétropropagation (ou Back Propagation en anglais) est un algorithme d'apprentissage supervisé utilisé pour entraîner des réseaux de neurones artificiels, notamment les Multilayer Perceptron (MLP). L'algorithme consiste à calculer la dérivée de l'erreur de sortie par rapport aux poids de chaque neurone dans le réseau, en utilisant la règle de la chaîne de dérivées. Cette dérivée est ensuite utilisée pour ajuster les poids des neurones, en utilisant une technique d'optimisation telle que la descente de gradient. L'objectif de l'algorithme est de minimiser l'erreur de sortie du réseau, en ajustant les poids de manière itérative jusqu'à atteindre une performance satisfaisante sur les données d'entraînement.
* Feed Forward : Feed Forward (ou propagation avant en français) est un concept clé dans les réseaux de neurones artificiels. Il désigne la propagation des entrées d'un réseau à travers les différentes couches de neurones, en direction de la couche de sortie, sans qu'il y ait de rétroaction des sorties vers les entrées. Cette propagation se fait en effectuant des calculs de pondération et de somme pondérée des entrées à travers chaque couche, suivis d'une fonction d'activation qui produit la sortie de chaque neurone. Les sorties des neurones de chaque couche sont ensuite utilisées comme entrées pour la couche suivante, jusqu'à atteindre la couche de sortie. Le Feed Forward est la phase d'inférence du réseau, où il prend des entrées et produit une sortie en utilisant les poids appris pendant l'apprentissage.
* Deep Learning : Le machine learning (ou apprentissage automatique en français) est une branche de l'intelligence artificielle qui permet aux ordinateurs d'apprendre à partir de données, sans être explicitement programmés pour chaque tâche. Il s'agit d'un processus d'apprentissage automatique où l'ordinateur apprend à partir de données d'exemples pour généraliser des modèles qui peuvent être utilisés pour faire des prédictions ou des classifications sur de nouvelles données. Les algorithmes de machine learning peuvent être supervisés, semi-supervisés ou non supervisés en fonction du niveau de supervision des données d'entraînement. Le machine learning est utilisé dans de nombreuses applications telles que la reconnaissance vocale, la détection de fraudes, la recommandation de produits et la reconnaissance d'images.
* Réseau neuronal convolutif (Convolutional Neural Network) : Un réseau neuronal convolutif (Convolutional Neural Network ou CNN en anglais) est un type de réseau de neurones artificiels conçu pour la reconnaissance d'images et la vision par ordinateur. Il est composé de plusieurs couches, y compris une ou plusieurs couches de convolution, qui permettent d'extraire automatiquement les caractéristiques des images. Les couches de convolution appliquent une série de filtres à l'image en entrée pour extraire des caractéristiques telles que les bords, les contours et les textures. Ces caractéristiques sont ensuite agrégées par des couches de pooling pour réduire la dimensionnalité et extraire des informations plus générales. Les couches entièrement connectées sont ensuite utilisées pour classifier les images en fonction des caractéristiques extraites. Les réseaux neuronaux convolutifs sont capables d'apprendre à partir de grandes quantités de données d'images et sont utilisés dans des applications telles que la reconnaissance faciale, la détection d'objets et la segmentation d'images.
* Recurrent neural network : Un réseau de neurones récurrents (Recurrent Neural Network ou RNN en anglais) est un type de réseau de neurones artificiels conçu pour traiter des données séquentielles telles que du texte, de la parole ou des séries temporelles. Contrairement aux réseaux de neurones classiques, les RNN ont des connexions récurrentes entre les neurones, ce qui leur permet de traiter des données séquentielles de manière itérative. Les sorties précédentes du réseau sont utilisées comme entrées pour les neurones suivants, permettant au réseau de mémoriser les informations du passé et de les utiliser pour influencer les prédictions futures. Les RNN sont couramment utilisés pour des tâches telles que la reconnaissance de la parole, la traduction automatique, la génération de texte et la prédiction de séries temporelles.
* TensorFlow : TensorFlow est une bibliothèque open source de traitement des données et de machine learning développée par Google. Il est conçu pour faciliter la création de réseaux de neurones artificiels et d'autres modèles de machine learning. TensorFlow permet aux utilisateurs de créer des graphes de calculs qui décrivent le flux de données entre les différentes couches d'un réseau de neurones, ce qui facilite l'entraînement et l'inférence de modèles. TensorFlow prend également en charge la distribution sur plusieurs processeurs et ordinateurs, ce qui permet aux utilisateurs de créer des modèles de machine learning à grande échelle. TensorFlow est utilisé dans de nombreuses applications, notamment la reconnaissance vocale, la classification d'images et la détection d'objets.
* Keras : Keras est une bibliothèque open source de réseaux de neurones artificiels écrite en Python. Elle permet aux utilisateurs de créer et d'entraîner des modèles de machine learning de manière simple et rapide. Keras offre une interface de programmation d'application (API) simple et cohérente pour construire des réseaux de neurones, avec une grande flexibilité pour personnaliser les architectures de réseau et les fonctions d'activation. Keras peut fonctionner sur différentes plates-formes de calculs, notamment TensorFlow, Theano et Microsoft Cognitive Toolkit. Keras est utilisé dans de nombreuses applications de machine learning, y compris la classification d'images, la reconnaissance vocale et la prédiction de séries temporelles. Keras est considéré comme l'un des meilleurs choix pour les débutants en apprentissage automatique en raison de sa facilité d'utilisation et de son approche conviviale pour la construction et l'entraînement de réseaux de neurones.
* Epoch : En apprentissage automatique, une époque (epoch en anglais) est une itération complète d'un ensemble de données à travers un modèle de machine learning pendant l'entraînement. Pendant chaque époque, le modèle passe par chaque exemple d'entraînement une fois, calcule les prédictions et les erreurs et ajuste les poids du réseau en fonction de ces erreurs à l'aide d'un algorithme d'optimisation. L'entraînement d'un modèle peut nécessiter plusieurs époques pour atteindre une précision suffisante, en particulier lorsque l'ensemble de données est grand. Les utilisateurs peuvent régler le nombre d'époques à entraîner pour un modèle, en fonction de la complexité du problème et de la quantité de données disponibles.
* batch\_size : En apprentissage automatique, la taille de lot (batch size en anglais) est le nombre d'échantillons de données utilisés pour mettre à jour les poids d'un modèle de machine learning à chaque itération. Lorsque les données sont trop volumineuses pour tenir en mémoire, elles sont divisées en lots plus petits qui peuvent être traités en parallèle. Les tailles de lots courantes sont 32, 64 et 128, bien que cela puisse varier en fonction de la taille de l'ensemble de données et de la capacité de traitement disponible. Les tailles de lots plus grandes peuvent permettre une convergence plus rapide, mais nécessitent également plus de mémoire et de puissance de traitement, tandis que les tailles de lots plus petites peuvent nécessiter plus d'itérations pour converger. Le choix de la taille de lot est donc un compromis entre la vitesse d'apprentissage et les contraintes de ressources.
* Fonction d’activation : En apprentissage automatique, une fonction d'activation est une fonction mathématique appliquée à la sortie d'une couche de neurones d'un réseau de neurones artificiels. Cette fonction est utilisée pour introduire de la non-linéarité dans le modèle, ce qui lui permet de capturer des motifs plus complexes dans les données. Les fonctions d'activation sont souvent des fonctions non linéaires telles que la fonction sigmoïde, la fonction ReLU (Rectified Linear Unit) ou la fonction tanh (tangente hyperbolique), qui sont appliquées élément par élément aux sorties de chaque neurone de la couche. La fonction d'activation détermine également la plage de valeurs possibles pour les sorties de la couche, ce qui peut avoir un impact sur la convergence du modèle pendant l'entraînement. Le choix de la fonction d'activation dépend du problème de classification ou de régression à résoudre et des caractéristiques de l'ensemble de données.
* ReLU : ReLU, ou Rectified Linear Unit, est une fonction d'activation non linéaire couramment utilisée dans les réseaux de neurones artificiels. Cette fonction est définie comme la sortie maximale entre 0 et la somme pondérée des entrées du neurone. En d'autres termes, si la somme pondérée des entrées est négative, la sortie est 0, sinon la sortie est égale à la somme pondérée. La fonction ReLU est souvent préférée à d'autres fonctions d'activation en raison de sa simplicité et de son efficacité dans la résolution de problèmes de classification et de régression. La fonction ReLU est également plus rapide à calculer que d'autres fonctions non linéaires, ce qui la rend idéale pour les réseaux de neurones profonds avec de nombreuses couches.
* Learning Rate (step) / Taux d’apprentissage : Le taux d'apprentissage, également appelé learning rate en anglais, est un hyperparamètre clé dans l'entraînement des réseaux de neurones. Il détermine la taille de chaque pas que l'algorithme d'optimisation prend lors de l'ajustement des poids du réseau en fonction des erreurs commises pendant l'entraînement. En d'autres termes, le taux d'apprentissage indique la rapidité avec laquelle le modèle doit ajuster ses prédictions en fonction des erreurs.
* Biais : En apprentissage automatique, le biais est un paramètre qui est ajouté à la somme pondérée des entrées d'un neurone dans un réseau de neurones artificiels. Le biais est un terme constant qui permet au réseau de neurones d'apprendre une fonction qui ne passe pas nécessairement par l'origine (0,0) et qui peut aider à mieux modéliser les données.

En d'autres termes, le biais représente la capacité d'un neurone à s'activer même si les entrées sont toutes nulles. Il est souvent ajouté à chaque couche de neurones et est ajusté pendant l'entraînement du réseau de neurones afin d'améliorer la précision des prédictions. Le biais est un paramètre important à ajuster lors de l'entraînement d'un réseau de neurones et son choix peut avoir un impact significatif sur la performance du modèle.

* Self Organised Map (SOM) : La carte auto-organisatrice (Self Organising Map ou SOM en anglais) est un type de réseau de neurones artificiels non supervisé qui peut être utilisé pour trouver des structures cachées dans des données multidimensionnelles. Le SOM est basé sur un réseau de neurones à deux dimensions, où chaque neurone est associé à un vecteur de poids et représente une région dans l'espace d'entrée.

L'entraînement de SOM consiste à présenter plusieurs exemples d'entrée au réseau de neurones, en ajustant les poids des neurones pour que les neurones voisins deviennent similaires pour les exemples d'entrée qui leur sont présentés. Les neurones qui sont similaires pour des exemples d'entrée similaires sont alors organisés dans des régions topologiquement similaires de la carte auto-organisatrice.

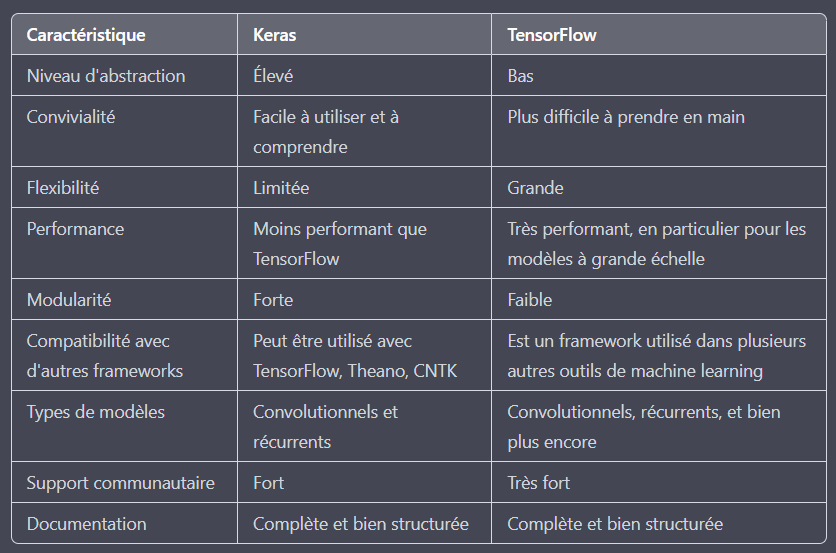
Les SOM sont souvent utilisées pour visualiser des données multidimensionnelles en deux dimensions, en attribuant chaque exemple d'entrée à son neurone le plus proche dans la carte auto-organisatrice. Les SOM peuvent également être utilisées pour la classification et la détection d'anomalies dans les données.

# Hypothèses :

* Un réseau de neurone peut être utilisé pour générer des films / vidéos par ex. à partir de texte donné par un utilisateur (Axel)
* Le Deep Learning est plus sensible à l’overfitting que les algorithmes non connexionnistes (Étienne)
* Dans le cadre des données Titanic, nous ne disposons pas de suffisamment de données pour faire un modèle performant avec le Deep Learning (Briand)
* Il existe des réseaux de neurones spécialisés (par ex. en computer vision ou en NLP). Càd des réseaux de neurones plus adaptés à certaines tâches (Aude)
* Chaque couche de réseau de neurone peut réaliser des fonctions différentes (Loïc)
* Pour faire de la classification ou de la régression, 3 couches (1 couche d’entrée et deux couches cachées), càd 3 vecteurs de poids, suffisent (Adrien)
* Le Deep Learning est plus utilisé que le Machine Learning (non connexionniste) (Osman)
* Le Deep Learning est plus approprié pour la reconnaissance d’images ou photos (computer vision). (Adeline)
* Il faut un environnement de travail spécialement dédié à l’entrainement d’un modèle de Deep Learning (Jean Paul)
* MLP et CNN sont utilisé pour différents types de problèmes (Tetyana)
* Le batch\_size consiste en le nombre d’itérations réalisé par le réseau de neurone pour ajuster les poids (Nicolas)
* Epoch consiste en un nombre d’itération réalisé par les réseaux de neurone pour ajuster les poids (Seydou)

# Plan d'action :

* Explorer les ressources
* Définir et comprendre les mots clefs
* Répondre à la problématique
* Vérifier les hypothèses
* Comparaison entre Keras et TensorFlow



* Faire les Workshop
* RER