

Introduction pratique au Deep Learning

L'objectif de ce TP est de se familiariser avec l'entraînement des réseaux de neurones et leur fonctionnement.

Les propriétés fondamentales et le fonctionnement des perceptrons multicouches sont retrouvés dans la majorité des modèles de Deep Learning.

L'application "Playground" de Tensorflow permet dans un environnement bac à sable de tester, expérimenter et assimiler ces mécanismes **sans aucun risque** : <https://playground.tensorflow.org/>

Ci-dessous, vous trouverez **quelques pistes d'exploration** à suivre ainsi que des indications sur **les fonctionnalités du Playground**.

Pistes d'exploration

Les pistes données dans cette partie sont données à titre indicatif. N'hésitez pas à dévier de celles-ci et expérimenter de votre propre initiative. A l'inverse, revenez vers celles-ci si nécessaire.

I. Choix des caractéristiques

Commencez par l'un des jeux de données simples (*Exclusive Or*, *Gaussian*) du problème type de Classification. Pour chaque modèle, réitérez plusieurs fois les entraînements et comparez les résultats obtenus.

1. Entraînez un premier modèle. Vous devriez pouvoir obtenir des modèles efficaces avec un minimum de neurones et de caractéristiques.
2. Réduisez le nombre de couches cachées à 1 avec une activation linéaire, vous obtiendrez le perceptron. Ici, nous l'entraînons avec la méthode de descente de gradient. Essayez les différentes données avec ce simple perceptron. Pensez à utiliser les formes de régularisation disponibles.
3. Toujours avec ce simple perceptron, essayez les différentes caractéristiques et hyperparamètres. Vous devriez pouvoir obtenir de bons résultats sur plusieurs bases de données.

II. Perceptron et données brutes

Pour cette partie, limitez-vous à l'usage des caractéristiques X1 et X2. Commencez à par un seul perceptron à activation linéaire sur les données *Circle*.

1. En augmentant le nombre de couches cachées de neurones à activation linéaire, observez les frontières générées.
2. Augmentez aussi le nombre neurones par couches.
3. Réessayez de trouver un modèle en utilisant une activation non linéaire. Observez à nouveau les frontières générées.
4. Testez les tailles de batch extrêmes et visualisez le comportement du modèle lors de l'apprentissage.
5. Testez les quantités de bruit extrêmes et visualisez le comportement du modèle lors de l'apprentissage.

III. Sélection du modèle approprié

Dans cette partie, nous nous concentrons sur le dataset "*Spiral*" qui nécessite des modèles plus complexes.

N'arrêtez pas vos entraînements prématurément. L'entraînement d'un modèle pour cette tâche peut nécessiter plusieurs centaines voire milliers d'epochs.

1. En appliquant les différents éléments vus précédemment, modifiez et faites converger ce modèle vers une solution satisfaisante.
2. Modifiez le learning rate pendant l'entraînement pour accélérer celui-ci.
3. Essayez d'entraîner un modèle le plus complexe possible (nombre de couches, nombre de features) et observez : le temps d'exécution d'une epoch, le temps nécessaire pour atteindre un modèle satisfaisant.

Description de l'interface

Bandeau horizontal

Vous trouverez à gauche des boutons permettant de :

- Démarrer, mettre en pause, réinitialiser un entraînement
- Exécuter l'entraînement epoch par epoch.

N'hésitez pas à mettre en pause un entraînement pour changer pendant celui-ci des hyperparamètres sans réinitialiser le modèle.

Le bandeau horizontal du haut comporte aussi des options permettant de modifier des hyperparamètres. On y trouve :

- Le learning rate
- La fonction d'activation
- Le type et le taux de régularisation (non abordés pour le moment)
- Le type de problème à optimiser

Si vous souhaitez faire converger le modèle plus vite, augmentez le learning. Attention cependant aux instabilités générées.

Section Data

Vous pouvez choisir entre plusieurs bases de données qui nécessiteront des modèles plus ou moins complexes.

Vous pouvez aussi modifier :

- Le ratio de données servant à l'entraînement et à l'évaluation
- La quantité de bruit introduit dans les données
- La taille des batchs

Changer le type de donnée réinitialise l'entraînement. Cependant si vous restez sur le même type de donnée, vous pouvez en générer de nouvelles sans réinitialiser le modèle. Cela peut s'assimiler à de l'augmentation de données.

Features

Par défaut, les coordonnées X1 et X2 sont utilisées mais vous pouvez aussi ajouter en entrée du modèle :

- Le carré de X1 ou X2
- Le produit de X1 par X2
- Le sinus de X1 ou X2

Ajoutez ou retirez des caractéristiques selon ce qui vous semble pertinent pour résoudre un problème. Cette étape nécessite une réflexion sur les données pour permettre une sélection de caractéristiques pertinentes pour la résolution du problème.

Les carrés représentent le domaine des données en entrée du bloc et la sortie est représentée en couleur avec les valeurs négatives en orange et les valeurs positives en bleu.

En passant la souris sur ces carrés, vous pourrez voir avec plus de précision les frontières générées.

Hidden Layers

Cette partie vous permet de modifier l'architecture du réseau.

Vous pouvez contrôler :

- Le nombre de couches cachées (profondeur du modèle)
 - Le nombre de neurones par couches cachées (largeur du modèle)
- Modifier le réseau peut permettre de déterminer des relations plus complexes entre les données.*

Vous pouvez visualiser les poids déterminés pour chaque neurone en regardant l'épaisseur et la couleur de chaque liaison.

En plus petit à gauche de chaque carré sont visibles les biais.

En survolant avec la souris ces éléments, vous pouvez voir les valeurs des poids et biais. Ces informations peuvent vous servir à déterminer quels éléments sont utiles ou non dans votre modèle. Cela peut aussi vous guider dans le choix des hyperparamètres.

Output

Dans cette section, vous trouverez divers indicateurs vous permettant de qualifier et quantifier la qualité du modèle et son apprentissage :

- Test loss, utilisée pour évaluer le modèle
- Train loss, que l'on cherche à minimiser par descente de gradient
- Deux courbes d'apprentissage représentant l'évolution de la train loss et de la test loss.
- Un espace représentant la prédiction faite par le modèle selon l'entrée. Par-dessus sont affichés les éléments de la base de données.

Il est possible de montrer les données de test ainsi que de discrétiser la sortie (utilisation d'un seuil à 0) afin d'obtenir une classification et de visualiser les frontières utilisées pour la classification