

TD6 – Réseaux de neurones pour la régression et la classification

L'objectif de ce TD est d'implémenter en *Python* un modèle de réseau de neurones. Deux versions seront développées, l'une pour un cas de régression et l'autre de classification. Des expérimentations seront ensuite menées pour évaluer la performance des modèles sur des jeux d'essai.

1. Présentation des éléments fournis

Plusieurs fichiers *Python* sont fournis afin d'aider à l'organisation du programme. La plupart d'entre eux sont à compléter afin de rendre le programme fonctionnel.

1. `rn_regression.py` : contient le programme principal permettant d'implémenter un réseau de neurones pour la régression
2. `lecture_donnees.py` : contient la fonction permettant de lire les données à partir d'un fichier texte
3. `normalisation.py` : contient la fonction permettant de normaliser les données
4. `decoupage_donnees.py` : contient la fonction permettant un découpage des données initiales en sous-ensembles d'apprentissage, de validation et de test.
5. `sigmoide.py` : contient la fonction permettant de calculer la valeur de la fonction sigmoïde et de sa dérivée
6. `lineaire.py` : contient la fonction permettant de calculer la valeur de la fonction linéaire et de sa dérivée
7. `relu.py` : contient la fonction permettant de calculer la valeur de la fonction relu et de sa dérivée
8. `calcul_cout_mse.py` : contient la fonction permettant de calculer la valeur de la fonction cout de type MSE (erreur quadratique)
9. `passe_avant.py` : contient la fonction permettant de réaliser une passe avant dans le réseau
10. `passe_arriere.py` : contient la fonction permettant de réaliser une passe arrière dans le réseau (rétropropagation)

2. Réseau de neurones pour la régression

Compléter les fichiers fournis pour rendre le programme fonctionnel. Il est conseillé de suivre l'ordre des fonctions indiqué dans la section précédente, et de tester le programme à chaque étape.

Afin de tester les fonctions développées, les fichiers utilisés dans les deux TD précédents pourront être utilisés : `food_truck.txt`, `houses.txt` et `notes.txt`.

3. Réseau de neurones pour la classification

En s'inspirant du programme contenu dans le fichier `rn_regression.py`, écrire dans un nouveau fichier `rn_classification.py` un programme permettant la mise en œuvre d'un réseau de neurones pour la classification. Comme vu dans le cours, une solution typique pour la classification est d'utiliser la fonction d'activation softmax pour la dernière couche du réseau et une fonction de coût de type entropie croisée.

Afin de tester les fonctions développées, le fichier `notes.txt` utilisé dans le TD précédent pourra être utilisé, ainsi que le célèbre jeu de données Iris contenant des mesures pour trois variétés de fleurs Iris (Setosa, Versicolour et Virginica) (fichier `iris.txt`).

4. Travail à rendre

Les deux programmes permettant la mise en œuvre d'un réseau de neurones pour la régression (rn_regression.py) et la classification (rn_classification.py) devront être finalisés et testés sur divers jeux de données de votre choix (par exemple à partir de : <https://archive-beta.ics.uci.edu/ml/datasets>) et avec différents paramétrages (nombre de couches cachées, nombre de neurones par couches cachées, types de fonction d'activation, ...) pour bien étudier le comportement des réseaux.

Ce travail sera à rendre pour le jeudi 10 novembre 2022. Vous devrez alors rédiger un compte-rendu électronique au format pdf indiquant votre démarche, vos résultats (tableaux de données et graphes), vos commentaires et vos interprétations des données. Ce rapport ainsi que votre programme *Python* devront être regroupés dans un fichier archive (.zip) qui sera déposé sur le site moodle.