

# Introduction au Deep Learning

## Présentation du module

L'**apprentissage profond**, ou **deep learning** est un ensemble de méthodes d'apprentissage automatique tentant de modéliser avec un haut niveau d'abstraction des données s'appuyant sur les réseaux de neurones artificiels. Cette approche vise à mimer l'apprentissage effectué par l'humain et est en perpétuelle évolution en accord avec notre compréhension de ces mécanismes.

Ces techniques d'apprentissage automatique ont permis des progrès importants et rapides dans les domaines de l'analyse de signaux (sonore, visuel, ...), donc de la vision par ordinateur (computer vision) et de la reconnaissance faciale, de la reconnaissance vocale et du traitement automatisé du langage.

Dans les années 2000, ces progrès ont suscité et grandement bénéficié des investissements privés, universitaires et publics importants, notamment de la part des GAFAM (Google, Apple, Facebook, Amazon, Microsoft) [Wikipédia et al.].

## Objectifs

L'objectif de ce module est de vous donner un premier aperçu du monde du deep learning et des outils pour pouvoir appréhender les bases de celui-ci. A la fin de ce module, vous serez capable de :

- Construire et implémenter un réseau de neurones,
- Utiliser keras pour implémenter un réseau de neurones et un réseau de neurones approfondi,
- Utiliser un réseau de neurones convolutifs (CNN).

## Pré-requis

- Utilisation de python
- Utilisation des librairies sklearn, pandas
- Notions de machine learning

## 1 : Introduction aux réseaux de neurones (1 jour)



### Démarche pédagogique

- Travail en autonomie
- Production individuelle

### Compétences

- Appréhender les réseaux de neurones
- Construire un réseau de neurones sans utiliser keras
- Construire un réseau de neurones en utilisant keras

### Objectif

1. Nous allons commencer par un peu de lecture sur les réseaux de neurones : lisez attentivement la ressource R1.1.
  - Y a-t-il des concepts que vous n'avez pas compris ? Notez-les au tableau. Les concepts seront vus ensemble avec le groupe.
2. Une fois que tous les concepts ont été bien expliqués, récupérez le jupyter notebook R1.2 (ML5\_perceptron\_J1.ipynb) et remplissez toutes les parties manquantes.
3. Après validation de votre jupyter notebook, vous pouvez passer maintenant à une implémentation du réseau de neurones à l'aide de la librairie keras.
  - Lisez attentivement la ressource R1.3
  - Appliquez le même fonctionnement sur la base de données

housepricedata.csv : implémentez le réseau de neurones en keras en vous aidant du notebook jupyter (ML5\_keras\_J1.ipynb).

- Partez de l'exemple donné dans la ressource R1.3 et donnez une première estimation de score de prédiction obtenu sur cette base de données.
- Jouez sur les paramètres de votre choix pour améliorer l'estimation du classifieur implémenté en vous aidant des ressources précédentes et R1.4, R1.5. Argumentez vos choix et si la prédiction peut être améliorée ou pas.

### Ressources

R1.1 <https://openclassrooms.com/fr/courses/4470406-utilisez-des-modeles-supervises-non-lineaires/4730716-entraenez-un-reseau-de-neurones-simple>

R1.2 Jupyter notebook (ML5\_method\_kit\_eleve\_J1.ipynb)

R1.3 <https://towardsdatascience.com/building-our-first-neural-network-in-keras-bdc8abbc17f5>

R1.4 <https://machinelearningmastery.com/how-to-choose-loss-functions-when-training-deep-learning-neural-networks/>

R1.5 <https://machinelearningmastery.com/grid-search-hyperparameters-deep-learning-models-python-keras/>

R1.6 <https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/>

Autre document intéressant : <https://www.v7labs.com/blog/deep-learning-guide>

### Livrables

- ☐ Le jupyter notebook de l'implémentation du perceptron
- ☐ Le jupyter notebook de l'implémentation du réseau de neurones avec keras