

## Présentation pédagogique

Cette UE est une introduction aux traitements des données et aux techniques d'apprentissage automatique.

## Prérequis

Bases de programmation en Python - Notion de variable aléatoire, fonction à plusieurs variables et notion de gradient.

## Acquis attendus à l'issue de l'UE

Maîtriser les concepts de base (théorie des probabilités, classification, régression, apprentissage non supervisé, fonction d'erreur, descente de gradient...) et connaître les outils du Machine Learning en python (NumPy, Pandas, PyTorch, Keras, Tensorflow, Scikit-learn, Seaborn) et les mettre en œuvre sur des exemples simples.

## Thèmes abordés

- Panorama du Machine Learning (types d'apprentissage, apport des réseaux de neurones profonds, génération de données, apprentissage de l'environnement...);
- Bases de statistiques pour le Machine Learning et boîte à outil du physicien : distributions de probabilité, notion d'échantillon de données, estimateurs statistiques et incertitudes, propagation des incertitudes, théorème de la limite centrale, variables indépendantes et corrélées, matrice de covariance, régression linéaire, ajustement par les moindres carrés et la méthode du maximum de vraisemblance ;
- L'apprentissage comme un problème de minimisation et techniques de minimisation d'une fonction à plusieurs variables (algorithme de descente de gradient). Minimisation sous contraintes (multiplicateur de Lagrange). Présentation des problèmes de régression et de classification. Présentation des algorithmes classiques (K-plus proches voisins, vector machines, Arbres de décision, approche par ensemble de classifieurs...);
- Structure d'un réseau de neurones, notion de couche, de fonction d'activation, notion de fonction d'erreur, algorithme d'entraînement d'un réseau de neurones par batchs. Introduction de la fonction SoftMax pour la multiclassification avec les réseaux de neurones. Notion d'optimiseur.
- Problématique de la dimensionalité des données, techniques de réduction et de préparation des données (retour vers une distribution centrée réduite, auto-encodeur).

## Savoir-faire techniques

Savoir coder un réseau de neurones (approche objet), la backpropagation utilisée pour son apprentissage en python (avec les structures de bases de NumPy ou PyTorch), mettre en place une régression multivariée, mettre en place une classification simple avec Scikit-learn.

# Organisation pédagogique

---

L'enseignement se fera sous la forme de Cours-TP intégré en salle machine.

# Ouvrages de référence

---

Le domaine étant en constante et rapide évolution et très neuf, les ouvrages sont peu durables. Ce cours nécessite un travail de veille d'une année sur l'autre pour rester pertinent.

# Informations pratiques

---

**Crédits**

6 ECTS

**Période d'enseignement :**

1<sup>er</sup> semestre de L2 (S3).

**Enseignement à distance :**

Non

**Enseignement en présentiel :**

Oui

**Volume horaire :**

CM : 12h

TD : 18h

TP : 40h

Travail personnel de l'étudiant : 12h

**Contact**

Bertrand LAFORGE