

Descriptif de module

Domaine : Economie & Services
Filière : Economie d'entreprise

1. Intitulé de module **Choix d'école Science des données** **2024-2025**

Code : 5731

Type de formation :

Bachelor Master MAS DAS CAS Autres :

Niveau :

- Module de base
 Module d'approfondissement
 Module avancé
 Module spécialisé
 Autres :

Caractéristique :

Module obligatoire dont l'échec peut entraîner l'exclusion définitive de la filière selon l'art. 14, du Règlement de filière du Bachelor of Science HES-SO en Economie d'entreprise

Type :

- Module principal
 Module lié au module principal
 Module facultatif ou complémentaire
 Autres :

Organisation temporelle :

- Module sur 1 semestre
 Module sur 2 semestres
 Semestre d'automne
 Semestre de printemps
 Autres : Modules sur 4 semestres pour le cursus En Emploi

2. Organisation

Crédits ECTS : 6

Langue principale d'enseignement :

- Français Italien
 Allemand Anglais
 Autres :

3. Prérequis

- Avoir validé le module
 Avoir suivi le module
 Pas de prérequis
 Autres :

4. Compétences visées / Objectifs généraux d'apprentissage

À l'issue du module, l'étudiant aura acquis une expérience pratique, ainsi qu'une base solide dans les trois domaines de compétences suivants.

- La gestion de projet
- La programmation : ce cours permet d'acquérir les fondements de la programmation, ainsi qu'une expérience pratique dans le langage de programmation Python.
- L'apprentissage automatique : les sujets abordés incluent les techniques supervisées et non supervisées, les algorithmes de base, les méthodes de validation et d'évaluation de modèle, l'interprétation critique des résultats et l'analyse de données.

A la fin de ce cours, les étudiant-e-s seront capables de :

Gestion de projet

- Expliquer la finalité de la gestion de projet et du management de projet
- Définir la spécificité et les caractéristiques d'un projet.
- Lister les besoins, le contexte et les objectifs de projet
- Identifier les compétences-clés du chef de projet.
- Comprendre le processus de documentation organisationnelle
- Gérer et évaluer les risques
- Maîtriser la planification et l'organisation de projet
- Vendre un projet au décideur

Programmation en Python

- Écrire et exécuter des instructions simples.
- Définir et manipuler les types de variables et collections de données.
- Résoudre des problèmes et des cas d'utilisations métiers à l'aide d'algorithmes et scripts.
- Créer des fonctions pour modulariser et/ou simplifier le code.
- Implémenter des structures conditionnelles et des boucles pour automatiser l'exécution répétée d'instructions.
- Importer et utiliser les bibliothèques classiques Python.

Apprentissage automatique

- Synthétiser des données de manière descriptives et visuelles.
- Identifier et formuler les problèmes qui relèvent de l'apprentissage automatique.
- Expliquer les concepts clés, algorithmes et modèles de l'apprentissage automatique.
- Sélectionner et appliquer des méthodes d'apprentissage appropriées pour des tâches spécifiques.
- Distinguer les ensembles de données, y compris leur structure, types et caractérisation.
- Appliquer des techniques de validation et d'évaluation des performances des modèles.
- Analyser et interpréter des résultats de manière critique.

5. Contenu et formes d'enseignement

Ce module est composé de trois composantes :

Gestion de projets

Contenu :

- Introduction à la gestion et au management de projet
- Structure du projet (plan de projet), besoins, contexte de projet
- Gestion des parties prenantes et plan de communication
- Capability Maturity Model
- Planification de projet, cycle de vie
- Structuration et estimation de projet (WBS, CBS)
- Ordonnancement (Gantt, Pert), RACI
- Référentiels métier de base (PMI, IPMA, Agile)
- Soft skills du chef de projet et pitch du projet

Forme d'enseignement :

- Cours basé sur la réalisation en équipe d'un plan de projet à partir d'un cas réel d'entreprise.
- Une période de cours sur les éléments théoriques incluant tests, discussions, et petits exercices corrigés en classe.
- Une période de coaching personnalisé des équipes de projet.

Programmation Python

Contenu :

- Introduction et démarrage avec Python : motivation, historique, caractéristiques et applications.
- Préparation d'un environnement de développement : installation, configuration, exploration de l'interface de développement (IDE), tel que Jupyter et VS Code.

- Introduction à la programmation : concepts fondamentaux de programmation, explication de la syntaxe (indentation), définition des variables, types de données, opérateurs et collections de données.
- Implémentation de la gestion des erreurs, des structures conditionnelles, et des boucles pour automatiser les instructions.
- Explication de notions de programmation avancées : fonctions, paramètres, gestion des valeurs retournées, portée des variables, expression lambda.
- Présentation des bibliothèques classiques Python pour la manipulation, l'analyse et la visualisation des données, tel que NumPy, Pandas, Matplotlib, Seaborn, etc.

Forme d'enseignement :

- Enseignement magistral, cours didactique.
- Exercices accompagnés, réalisés de manière autonome et corrigés en classe.

Apprentissage automatique

Contenu :

- Visualisation des données : variables quantitatives (continues ou discrètes) vs qualitatives (nominales ou ordinales), nuages de points, histogrammes, diagrammes en boîtes (boxplots), cartes de chaleur, PCA pour la visualisation, etc.
- Statistiques descriptives : mesures de tendance centrale (moyenne, médiane, mode), mesures de dispersion (variance, IQR), corrélation, etc.
- Introduction à l'apprentissage automatique : définitions, histoire, applications, classification des problèmes, apprentissage supervisé (régression et classification) et non supervisé (réduction de la dimensionalité et clustering).
- Apprentissage supervisé : régression linéaire, régression logistique, arbres de décision, SVM, méthodes d'ensembles (boosting et bagging), réseaux de neurones, etc.
- Apprentissage non supervisé : PCA, K-means, clustering hiérarchique.
- Sélection de modèles : critères de performance (RMSE, MAE, matrice de confusion, précision, rappel, courbes ROC, F-score, etc.) et techniques de validation (validation croisée, hold-out).
- Analyse et critique des modèles : interprétation des résultats, identification des biais, robustesse, généralisation, surajustement.

Forme d'enseignement :

- Enseignement magistral, cours didactique.
- Exercices accompagnés, réalisés de manière autonome et corrigés en classe.

6. Utilisation de l'Intelligence Artificielle

Enseignement

Dans le cadre de l'enseignement, nous adoptons une approche pédagogique qui vise à développer les compétences fondamentales et le raisonnement critique des étudiants sans dépendance excessive aux outils d'intelligence artificielle (IA) générative.

Dans la partie gestion de projets, l'accent est mis sur l'apprentissage des compétences organisationnelles et managériales sans l'assistance directe des outils d'IA. Cela permet aux étudiants de développer une base solide en gestion de projet.

L'utilisation d'outils d'IA pour les parties programmation et apprentissage automatique est également fortement déconseillée. Cette mesure garantit que les étudiants acquièrent une compréhension authentique des concepts et développent une capacité à les appliquer, sans s'appuyer sur des solutions automatisées. Cependant, l'utilisation de tels outils sera abordée en fin de cours afin d'illustrer son potentiel en matière de production de code et d'analyse de données.

Évaluation

Dans le cadre de l'évaluation, l'utilisation de l'intelligence artificielle est formellement interdite lors des contrôles continus. Cette interdiction vise à garantir l'évaluation équitable des compétences individuelles sans le soutien d'outils d'IA.

Cependant, dans le cadre de la réalisation du dossier en groupe (gestion de projet), l'utilisation des outils d'IA est tolérée, sous réserve que les étudiants cherchent de l'information nécessaire à comprendre le contexte et qu'ils puissent clairement identifier les parties de leur travail assistées par IA, justifier leur utilisation dans leurs analyses, ainsi que comprendre les limites de ces outils.

7. Modalités d'évaluation et de validation

L'évaluation de chaque cours de fera de la manière suivante :

Ce module sera évalué en contrôle continu durant le semestre académique. La note pour la composante Gestion de projet sera à 50% en individuel et 50% en collectif (dossier et pitch). Les composantes programmation et apprentissage automatique seront à 100% évaluées en individuel.

La note finale du module est obtenue en faisant la moyenne des trois composantes. Chaque composante a une pondération égale aux autres, soit 1/3 de la note pour Gestion de projet, 1/3 pour Programmation et 1/3 pour Apprentissage automatique.

8. Modalités de remédiation

- Remédiation obligatoire si le résultat du module est compris entre 3,5 et 3,9 / 6.
- Pas de remédiation
- Autres modalités :

Autres modalités de remédiation

En cas de remédiation, seule la note de l'examen de remédiation sera prise en compte.

Un module répété ne peut pas être remédié.

9. Bibliographie

Gestion de projet

- La conduite de projet, Thierry Hougron (3^{ème} édition), Les 126 règles pour piloter vos projets avec succès, Dunod, 2015.
- Projet Management Institute, A Guide to the project management body of knowledge (PMBOK Guide), 7^{ème} éd, 2021.
- Project Management by ICB4, Van Haren Publishing; 1^{ère} édition, 2023
- Jérôme Maes, François Debois, La boîte à outils du chef de projet, 3^{ème} édition, Dunod, 2023
- Le Manifeste Agile : <https://agilemanifesto.org/>
- Le Guide Scrum : <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-French.pdf>
- Agile vidéos : <https://www.scrumlife.tv> ; <https://www.youtube.com/@ScrumLife/videos> ; <https://youtube.com/playlist?list=PLRECYZitnonhv14xGf2wHu-KPABdojO&si=9h8orjzlrAMssgDh>

Programmation et apprentissage automatique

- Python. <https://www.python.org/>
- Anaconda (Individual Edition). <https://www.anaconda.com/products/individual>
- Jupyter Lab. <https://jupyterlab.readthedocs.io/en/latest/>
- W3Schools (Python Tutorial) <https://www.w3schools.com/python/>
- Pandas (Python Data Analysis Library, <https://pandas.pydata.org/>)
- Numpy (Fundamental package for scientific computing with Python, <https://numpy.org>)
- Matplotlib (Visualization with Python, <https://matplotlib.org>)
- Seaborn (Statistical data visualization, <https://seaborn.pydata.org/>)
- Scipy (Scientific computing with Python, <https://scipy.org/>)
- Scikit-learn (Machine learning in Python, <https://scikit-learn.org/stable/>)