

Année universitaire 2025/2026

# Informatique, Décision, Données - 1re année de Master

**Responsable pédagogique** : Hassan AISSI

**Crédits ECTS** : 60

## LES OBJECTIFS DE LA FORMATION

La **1re année du Master Informatique parcours Informatique, Décision et Données** met l'accent sur les aspects fondamentaux de l'informatique, sur les mathématiques discrètes, les sciences des données, la recherche opérationnelle et l'aide à la décision. Cette première année commune pour les étudiantes et les étudiants souhaitant s'orienter vers les 2e années de Master suivantes : Modélisation, Optimisation, Décision et Organisation (MODO) Intelligence Artificielle et Sciences des Données (IASD). Tous les cours seront dispensés en anglais dès lors qu'au moins un(e) étudiant(e) est anglophone.

### Les objectifs de la formation :

- Assimiler la modélisation de systèmes et de situations pour l'aide à la décision dans les organisations ;
- Utiliser et adapter les méthodes de résolution ;
- Analyser et exploiter des résultats ;
- Utiliser le déploiement d'outils dans les organisations.

## MODALITÉS D'ENSEIGNEMENT

**Les Modalités des Contrôles de Connaissances (MCC) détaillées sont communiquées en début d'année.** La formation débute la dernière semaine d'août et la présence en cours est obligatoire. Les enseignements de la première année du Master mention Informatique parcours Informatique, Décision et Données sont organisés en semestre 1 et 2. Chaque semestre est constitué d'un bloc fondamental et d'UE complémentaires auxquels s'ajoute un bloc stage pour le semestre 2. À chaque UE est associé un certain nombre de crédits européens (ECTS) à chaque semestre est associée la somme des ECTS associés aux UE composant le semestre.

## ADMISSIONS

Titulaires d'un diplôme BAC+3 (180 crédits ECTS) ou équivalent à Dauphine, d'une université, d'une école de commerce, d'ingénieurs ou d'un autre établissement de l'enseignement supérieur dans les domaines suivants : informatique, mathématiques, gestion ou économie (avec options recherche opérationnelle ou techniques quantitatives ou informatique).

## PROGRAMME DE LA FORMATION

- Semestre 1
  - Bloc fondamental 1
    - [Algorithmic and advanced programming](#)
    - [Anglais 1](#)
    - [Artificial Intelligence](#)
    - [Graph theory](#)
    - [Mathematics for data science](#)
  - UE Complémentaires
    - [Décision collective, décision multicritère](#)
    - [Game theory](#)
    - [Programmation objet avancée](#)
    - [Système et algorithmes répartis-SAR](#)
- Semestre 2

- Bloc fondamental 2
  - [Anglais 2](#)
  - [Combinatorial optimization](#)
  - [Computer ethics & data protection](#)
  - [Data base management system](#)
  - [Machine Learning](#)
- UE Complémentaires
  - [Decision under uncertainty](#)
  - [Logic theory](#)
  - [ML Project/Data science](#)
- Bloc stage
  - [Stage](#)

## DESCRIPTION DE CHAQUE ENSEIGNEMENT

### SEMESTRE 1

---

#### Bloc fondamental 1

## Algorithmic and advanced programming

ECTS : 5

**Enseignant responsable** : ROLAND GRAPPE (<https://dauphine.psl.eu/recherche/cvtheque/grappe-roland>)

**Langue du cours** : Anglais

**Volume horaire** : 36

#### Description du contenu de l'enseignement :

Explore advanced algorithmic design in the intensive course, Algorithms and Advanced Programming. Participants delve into cutting-edge concepts, optimization strategies, and intricate problem-solving methods, fostering a deep understanding of complex computational challenges. The program aims to equip individuals with the skills necessary to navigate the dynamic landscape of algorithmic innovation, making it a valuable experience for those seeking mastery in advanced computer science. Here is a possible list of contents, which might change according to the current trends or the lecturer's inclinations.

- Greedy Algorithms and Matroids
- Dynamic Programming
- Graph Algorithms I: Efficient Problem Solving
- Graph Algorithms II: Complex Problem Challenges
- Computational Geometry
- Stable Marriage
- Introduction to Integer Programming

#### Compétences à acquérir :

Develop proficiency in designing and implementing advanced algorithms

#### Pré-requis recommandés

- knowledge of Python
- knowledge of basic algorithms (sorting, shortest paths algorithms, minimum spanning tree algorithms...)

#### Mode de contrôle des connaissances :

- A programming project during the semester
- A final exam on paper

#### Bibliographie, lectures recommandées :

- Introduction to Algorithms, third edition, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. MIT Press.
- Combinatorial Optimization, William Cook, William Cunningham, William Pulleyblank, Alexander Schrijver. Wiley.

---

## Anglais 1

ECTS : 2

**Enseignant responsable** : CATHERINE BOILLOT-PATTERSON (<https://dauphine.psl.eu/recherche/cvtheque/boillot-patterson-catherine>)

**Langue du cours** : Anglais

**Volume horaire** : 19.5

#### Description du contenu de l'enseignement :

Expression oral et écrite: anglais des affaires, faire un compte rendu oral en public, rédiger lettres et rapports etc  
Familiariser les étudiants avec la terminologie économique à partir de thèmes d'actualités

#### Compétences à acquérir :

Fournir aux étudiants les outils linguistiques nécessaires pour fonctionner efficacement l'entreprise et avec leurs partenaires.

**Mode de contrôle des connaissances :**

Contrôle continu

---

# Artificial Intelligence

ECTS : 3

**Enseignant responsable :** JULIEN LESCA (<https://sites.google.com/site/julienlesca/home>)

**Langue du cours :** Anglais

**Volume horaire :** 36

**Description du contenu de l'enseignement :**

This class introduces the main ideas and algorithms that let an artificial agent plan and carry out sequences of actions to reach a goal. We explore several families of techniques:

- **Search algorithms:**  
How an agent explores a state space. We look at uninformed methods like breadth-first and depth-first search, and informed methods such as greedy best-first search and A\*, which use heuristics to guide the search.
- **Local search and optimization:**  
Techniques for improving a solution step by step, including hill climbing, simulated annealing, local beam search, and genetic algorithms.
- **Constraint satisfaction problems (CSPs):**  
A framework for modelling problems using variables and constraints. We cover AC-3 for constraint propagation and backtracking search for finding consistent assignments.
- **Nondeterministic and partially observable environments:**  
How agents plan when actions have uncertain effects or when they cannot see the full state of the world. We introduce AND-OR tree search and belief states.
- **Multi-agent environments:**  
Basic ideas from game playing, including minimax and alpha-beta pruning, where agents must reason about opponents.
- **Classical planning:**  
An introduction to PDDL and planning-graph techniques for encoding and solving high-level planning tasks.

**Compétences à acquérir :**

By the end of this course, students will be able to:

- **Model a variety of decision and planning problems** using appropriate representations (state spaces, constraints, planning formalisms, etc.).
- **Understand and select suitable algorithmic approaches**—such as search, optimization, or constraint-based methods—to compute solutions.
- **Use existing algorithms and solvers effectively**, and assess their strengths, limitations, and suitability for a given problem.

**Mode de contrôle des connaissances :**

50% Project - 50% Exam

**Bibliographie, lectures recommandées :**

Russell, S., & Norvig, P. (2009). Artificial Intelligence: A Modern Approach (3rd ed.). Upper Saddle River, NJ: Prentice Hall.

En savoir plus sur le cours : <https://sites.google.com/site/julienlesca/teaching/ai>

---

# Graph theory

ECTS : 5

Enseignant responsable : MICHAÏL LAMPIS (<https://www.lamsade.dauphine.fr/~mlampis/>)

Langue du cours : Anglais

Volume horaire : 36

## Description du contenu de l'enseignement :

Volume horaire :

CM : 18h

TD : 18h

This class provides an introduction to Graph Theory from a computer science and algorithms perspective. Our goal will be to understand which optimization problems on graphs admit efficient (polynomial-time) algorithms and how graph structure affects algorithmic tractability.

In the first half of the course, we will cover basic topics in graph theory, including problems related to connectivity and cuts (Menger's theorem), Coloring (Brooks' theorem), and Bipartite matching (Konig's theorem). The second half will focus on studying the interplay between computational complexity and graph structure by considering notable classes of graphs, including planar, chordal, interval, and split graphs. The objective will be to understand how such graph structures can arise in practice and how they can be exploited algorithmically.

## Compétences à acquérir :

At the end of the course students will be familiar with the most important optimization problems on graphs and be able to distinguish several notable cases where these problems are efficiently solvable.

## Pré-requis recommandés

This class can be considered as a natural continuation of the "Graph Algorithms" course offered on the L3 level. It is recommended for students to be familiar with the contents of this (or a similar) previous course, including the basic definitions of graphs and fundamental algorithms (such as BFS, DFS). However, this course is intended to be self-contained and familiarity with these topics, though helpful, is not mandatory.

---

# Mathematics for data science

ECTS : 5

Enseignant responsable : Clément ROYER (<https://www.lamsade.dauphine.fr/~croyer/cours.html>)

Langue du cours : Anglais

Volume horaire : 36

## Description du contenu de l'enseignement :

Volume horaire :

CM : 18h

TD : 18h

Data science relies heavily on mathematical concepts from analysis, linear algebra and statistics. In this course, we will investigate the theoretical foundations of data science, through two axes. The first part of the course will focus on (convex) optimization problems. Optimization is indeed at the heart of the key advances in machine learning, as it provides a framework in which data science tasks can be modeled and solved. The second part of the course will be concerned with statistical tools for data science, that are instrumental in studying the underlying distribution of data. We will cover statistical estimation in connection with regression tasks, as well as concentration inequalities for random vectors and random matrices.

## Compétences à acquérir :

- Identify and exploit convexity for sets, functions, and optimization problems.
- Derive optimality and duality results for convex optimization formulations.
- Analyze properties of statistical estimators according to the data at hand.
- Apply concentration inequalities to random vectors and matrices.

### Pré-requis obligatoires

Basic knowledge of linear algebra and real analysis.

### Bibliographie, lectures recommandées :

References: S. Boyd et L. Vandenberghe, Convex optimization (2004) M. Mahoney, J. C. Duchi, A. C. Gilbert (eds), The mathematics of data (2018) J. A. Tropp, An introduction to matrix concentration inequalities (2015)

En savoir plus sur le cours : <https://www.lamsade.dauphine.fr/~croyer/teachMDS.html>

---

## UE Complémentaires

# Décision collective, décision multicritère

ECTS : 5

Enseignant responsable : Hassan AISSI

Langue du cours : Français

Volume horaire : 36

### Description du contenu de l'enseignement :

Volume horaire : CM : 18h TD : 18h Introduction au processus de décision Introduction à la théorie du choix social Procédures d'agrégation multicritères de type critère de synthèse Procédures d'agrégation multicritères de type relation de surclassement, Illustration des méthodes sur des cas réels

### Compétences à acquérir :

Présenter les principales familles de méthodes d'agrégation multicritère existantes et mise en œuvre de telles méthodes dans des situations réelles de décision.

---

# Game theory

ECTS : 5

Enseignant responsable : ARARAT HARUTYUNYAN (<https://www.lamsade.dauphine.fr/~aharutyunyan/>)

Langue du cours : Anglais

Volume horaire : 36

### Description du contenu de l'enseignement :

Volume horaire :

CM : 18h

TD : 18h

We study some basic models and results on non-cooperative game theory. The course is divided in five parts:

1. Introduction to game theory: Notations and basic concepts
2. Zero-sum games: Theorem of Von Neumann
3. Nash equilibrium, their existence and mixed strategies; Nash's Theorem
4. Combinatorial Games, Game trees, Zermelo's Theorem
5. Topology: Sperner's Lemma and Applications

### Compétences à acquérir :

The student will know important concepts of game theory, and learn how to solve problems and prove theorems: Nash Equilibrium, Von Neumann's Theorem, Zermelo's Theorem, Combinatorial games, etc.

### Pré-requis obligatoires

A first course in probability theory; a first course in linear algebra (good familiarity of convex sets); basic notions of graph theory/ algorithmic theory;

### Mode de contrôle des connaissances :

CC (midterm) + Final Exam

### Bibliographie, lectures recommandées :

# Programmation objet avancée

ECTS : 5

**Enseignant responsable** : KHADOUJA ZELLAMA (<https://dauphine.psl.eu/recherche/cvtheque/zellama-khaddouja>)

**Langue du cours** : Français

**Volume horaire** : 36

## Description du contenu de l'enseignement :

Volume horaire :

CM : 18h

TD : 13h30

TP : 4h30

1. Introduction à la programmation objets avancée: Rappels des concepts classes, héritage, interface, exceptions.
2. Définition et utilisations des classes abstraites et interfaces
3. Application au traitement des collections.
4. Types paramétrés et wildcards.
5. Traitement des processus (thread).
6. Introduction à la programmation de tâches s'exécutant en parallèle.
7. Gestion de ressources partagées entre processus.
8. Modélisation objet des systèmes d'information en utilisant les design patterns.
9. Conception et gestion de bases de données relationnelles et développement d'une application JAVA pour MAJ la BD.

## Compétences à acquérir :

- Consolider les bases en programmation objets et particulièrement en langage JAVA 8 et plus.
- Faire de la programmation concurrente et parallèle (multi-threading).
- Modélisation de système d'information en utilisant les design patterns pour résoudre des problèmes récurrents.
- Définir les spécifications fonctionnelles d'un système d'information et savoir les automatiser à travers une application JAVA.
- Modéliser une base de données relationnelle et la gérer à travers une application JAVA.

## Pré-requis obligatoires

Maîtriser les concepts de classes et objets ainsi que les modèles UML (diagramme de classe, cas d'utilisation, objets).

## Pré-requis recommandés

Avoir quelques connaissances en langage orienté objets comme C++, Python ou autre langage objet.

## Mode de contrôle des connaissances :

Projet individuel (Pr) à réaliser chez soi pour une durée d'un mois ou un mois et demi. Ce projet consiste à automatiser certaines fonctionnalités d'un système d'information proposé et de gérer toutes les données relatives à ce dernier à travers une BD.

Une note de contrôle continu (CC) individuelle à préciser en début de cours.

Un examen sur table individuel (E).

Note d'enseignement =  $0,1CC + 0,3Pr + 0,6E$

## Bibliographie, lectures recommandées :

Programmer en Java 6e Edition - C. Delannoy. Java in a nutshell - D. Flanagan. Thinking in Java - B. Eckel. Programmation concurrente en Java - B. Goetz. Tête la première, Design Patterns - E. Freeman et al.

**ECTS** : 5

**Enseignant responsable** : Joyce **EL HADDAD**

**Langue du cours** : Français

**Volume horaire** : 36

**Description du contenu de l'enseignement :**

Volume horaire : CM : 18h TD : 18h Les applications réparties s'exécutent sur un ensemble de machines connectées en réseau. Elles représentent un ensemble de composants qui coopèrent pour réaliser un objectif commun en utilisant le réseau comme un moyen d'échanger des données. Ce cours vise à présenter les concepts élémentaires des systèmes et les algorithmes associés aux environnements répartis Introduction aux systèmes répartis et à l'algorithmique répartie. Présentation du modèle de répartition basé sur les échanges de messages. Présentation des concepts liés à la communication : contrôle de flux, synchronisation de processus, relation de causalité, réseaux FIFO. Présentation des concepts liés au temps et à la concurrence : horloges logiques, exclusion mutuelle.

**Compétences à acquérir :**

Introduction aux systèmes répartis.

**Pré-requis recommandés**

- Systèmes d'exploitation
- Réseaux

**Bibliographie, lectures recommandées :**

- Systèmes d'exploitation, systèmes centralisés, systèmes distribués, A. Tanenbaum, Dunod-Prentice Hall, 1994 - La communication et le temps dans les réseaux et les systèmes répartis, M. Raynal, Collection Direction des Etudes et des Recherches d'EDF n°75, Hermès, 1991

En savoir plus sur le cours : <https://joyceelhaddad.github.io/non-menu-page/>

---

## SEMESTRE 2

---

### Bloc fondamental 2

## Anglais 2

**ECTS** : 2

**Enseignant responsable** : CATHERINE BOILLOT-PATTERSON (<https://dauphine.psl.eu/recherche/cvtheque/boillot-patterson-catherine>)

**Langue du cours** : Anglais

**Volume horaire** : 18

**Description du contenu de l'enseignement :**

Fournir aux étudiants les outils linguistiques nécessaires pour fonctionner efficacement dans l'entreprise et avec leurs partenaires européens. Expression orale / écrite : anglais des affaires, faire un compte rendu oral en public, rédiger, lettres, rapports, résumé de conférences, réunions. Préparation au TOEIC : Test of English for International Communication. Traduction économique : Familiariser les étudiants avec la terminologie économique à partir de thèmes d'actualité. Travail en laboratoire et/ou en salle audiovisuelle à partir de documents authentiques.

**Compétences à acquérir :**

Fournir aux étudiants les outils linguistiques nécessaires pour fonctionner efficacement dans l'entreprise et avec leurs partenaires européens.

---

## Combinatorial optimization

**ECTS** : 5

**Enseignants** : DENIS CORNAZ, ARARAT HARUTYUNYAN  
<https://www.lamsade.dauphine.fr/~cornaz/>  
<https://www.lamsade.dauphine.fr/~aharutyunyan/>

**Langue du cours** : Anglais

**Volume horaire** : 39

**Description du contenu de l'enseignement** :

We start the course with the geometry of polyhedra and polytopes. Next we will study the matching polytope in bipartite graphs, and then in general graphs.

**Compétences à acquérir** :

Theoretical foundation of polyhedral combinatorics

**Bibliographie, lectures recommandées** :

Reference text: Alexander Schrijver, "A course in Combinatorial Optimisation" available at <https://homepages.cwi.nl/~lex/files/dict.pdf>

## Computer ethics & data protection

**ECTS** : 1

**Enseignant responsable** : THIERRY JAILLET (<https://dauphine.psl.eu/recherche/cvtheque/jaillet-thierry>)

**Langue du cours** : Anglais

**Volume horaire** : 9

**Description du contenu de l'enseignement** :

Presentation of ethical problem situations or dilemmas regarding information systems and the processing of personal data. Reminder of national and European GDPR and AI Act regulations. Team reflection on ethical issues in professional situations.

**Compétences à acquérir** :

To be an IT professional aware of the ethical issues induced by the processing of personal data, the use of AI and the social and societal responsibility of information systems managers

## Data base management system

**ECTS** : 5

**Langue du cours** : Anglais

**Volume horaire** : 36

**Compétences à acquérir** :

Savoir gérer les données, qu'il s'agisse de données sur le Web, de données collectées à partir de différents silos des entreprises, ou de données traitées par des laboratoires scientifiques (par exemple dans le cadre de la bio-informatique, les sciences de la Terre, la sociologie, l'économétrie, etc.) Couvrir les techniques internes des systèmes de gestion de base de données (SGBD) qui sont responsables de l'optimisation de l'évaluation de requêtes SQL.

## Machine Learning

**ECTS** : 5

**Enseignant responsable** : YANN CHEVALEYRE (<https://dauphine.psl.eu/recherche/cvtheque/chevaleyre-yann>)

**Langue du cours** : Français

**Volume horaire** : 36

**Description du contenu de l'enseignement** :

Volume horaire : CM : 18h TD : 18h

1. Introduction
  1. What is Machine Learning
  2. A simple method: k-nearest neighbors

3. Evaluation of classifiers
4. Maximum Likelihood and Maximum A posteriori
2. Generative Learning
  1. Maximizing the Likelihood of the examples
  2. Linear Discriminant Analysis and Naive Bayes
3. Discriminative Learning
  1. Maximizing the likelihood and the a posteriori probability of labels
  2. Logistic Regression
  3. Stochastic gradient descent (SGD)
  4. SGD for generalized linear models
  5. Beyond linearity: kernelization of the SGD
4. Unsupervised Learning
  1. Learning latent models: the Expectation-Maximization Algorithm
  2. clustering: k-means, DBSCAN
  3. Learning probability density functions: mixtures of gaussians
5. Introduction to Bayesian Learning
  1. Bayesian Linear Regression
  2. Laplace method
6. Introduction to Neural Networks

**Compétences à acquérir :**

Understand most useful machine learning algorithms

**Pré-requis recommandés**

- Connaissances de base en statistiques et algèbre linéaire

**Mode de contrôle des connaissances :**

CC+Examen

**Bibliographie, lectures recommandées :**

- Friedman, Tibshirani, Hastie. The Elements of Statistical Learning - Chloé Azencott. Introduction au Machine Learning - Cornuéjols, Miclet. Apprentissage artificiel: Concepts et algorithmes

**UE Complémentaires**

## Decision under uncertainty

ECTS : 5

**Enseignant responsable :** HUGO GILBERT (<https://dauphine.psl.eu/recherche/cvtheque/gilbert-hugo>)

**Langue du cours :** Anglais

**Volume horaire :** 36

**Description du contenu de l'enseignement :**

\* Complete uncertainty (non-probabilistic): definitions of decision in uncertainty, presentation of classical criteria for decision in uncertainty (MaxMin, Min Max Regret, Hurwicz, Laplace, etc.) \* Sequential Decision (Probabilistic Uncertainty): EU Model, Decision Trees (Trees Containing Decision Nodes, Random Nodes, and Terminal Nodes), and solving these problems by dynamic programming \* Introduction to non-EU, RDEU and CEU models. \* Elicitation/Learning of decision-making models. \* Markov Decision Process. \* Probability, Independence, Bayes' rule. \* Reasoning in Bayesian Networks (Exact and Approximate Inferences). ----- \* Incertain complet (non probabiliste) : définitions de la décision dans l'incertain, présentation des critères classiques de décision dans l'incertain (MaxMin, Min Max Regret, Hurwicz, Laplace, etc.) \* Décision séquentielle (incertain probabiliste) : modèle EU, arbres de décision (arbres contenant des nœuds décision, nœuds hasard et nœuds terminaux), et la résolution de ces problèmes par programmation dynamique \* Introduction aux modèles non-EU, RDEU et CEU. \* Elicitation et apprentissage des modèles décisionnels \* Processus de décision markovien \* Rappel de probabilité, indépendance, règle de Bayes \* Raisonnement dans les Réseaux Bayésiens (inférences exactes et approchées)

**Compétences à acquérir :**

Introduce students to the main techniques of decision-making under uncertainty. Présenter aux étudiantes et étudiants les techniques principales de la décision dans l'incertain.

### Pré-requis recommandés

An introduction to decision theory.

### Mode de contrôle des connaissances :

The course is evaluated by a written exam as well as a programming project.

### Bibliographie, lectures recommandées :

\* von Neumann, John and Oskar Morgenstern, Theory of Games and Economic Behaviour, Princeton University Press, 1947. \* Savage, Leonard J., The Foundations of Statistics, Dover, 1954. \* Puppe, C., Distorted probabilities and choice under risk (Vol. 363). Springer Science & Business Media, 1991. \* Barbera, S., Hammond, P.J., & Seidl, C. Editors, Handbook of Utility Theory: Volume 2: Extensions. Springer Science & Business Media, 1998. \* Barbera, S., Hammond, P.J., & Seidl, C. Editors, Handbook of Utility Theory: Volume 1: Principles. Springer Science & Business Media, 1998.

---

## Logic theory

ECTS : 5

Enseignant responsable : GABRIELLA PIGOZZI (<https://dauphine.psl.eu/recherche/cvtheque/pigozzi-gabriella>)

Langue du cours : Anglais

Volume horaire : 36

### Description du contenu de l'enseignement :

- Revision of fundamental concepts of classical logic (propositional and first-order logic) that will be necessary for this course
- Proofs of soundness and completeness of propositional and first-order logic
- Decidability of propositional logic
- Undecidability of first-order logic
- Gödel's incompleteness theorems (without proofs)
- Modal logic: main systems and proofs using tableaux method
- Formal verification by model-checking:
  - Linear-time Temporal Logic (LTL)
  - Computation Tree Logic (CTL)

### Compétences à acquérir :

In the first part of the course, students will learn some key results in classical logic and logical metatheory: soundness, completeness and (un)decidability of classical logic.

The second part of the course will focus on some non-classical logics, namely modal logic and temporal logics. Temporal logics will be used in formal verification by model-checking (the process to verify the correctness of computer systems concerning some specified behaviour).

### Pré-requis recommandés

Classical logic (propositional and first-order)

### Mode de contrôle des connaissances :

Written exam

---

## ML Project/Data science

ECTS : 5

Enseignant responsable : PIERRE WOLINSKI (<https://dauphine.psl.eu/recherche/cvtheque/wolinski-pierre>)

Langue du cours : Anglais

Volume horaire : 36

### Description du contenu de l'enseignement :

Project: work by groups of two, build or use an existing (real) dataset, process the dataset by using the methods presented in the courses.

Implement the different methods presented in the Machine Learning course by using the libraries NumPy, Pandas and scikit-learn on real or synthetic data:

- k-nearest neighbors (k-NNs);
- Linear/Quadratic Discriminant Analysis (LDA/QDA);
- Logistic regression;
- Clustering: k-means, agglomerative clustering;
- Decision tree learning, random forests;
- Neural networks.

**Compétences à acquérir :**

Implement common Machine Learning techniques in Python (with the help of the libraries NumPy, Pandas and scikit learn).

**Pré-requis obligatoires**

Programming language: Python.

**Mode de contrôle des connaissances :**

Graded tutorial (TP noté) + Project.

---

**Bloc stage**

## Stage

ECTS : 4

Langue du cours : Français

**Description du contenu de l'enseignement :**

Stage obligatoire d'un minimum de 3 mois.

---

**Document susceptible de mise à jour - 02/04/2026**

**Université Paris Dauphine - PSL - Place du Maréchal de Lattre de Tassigny - 75775 PARIS Cedex 16**