Course catalogue

Courses on the Metz campus during the fall semester

Fall Semester

Metz Campus of CentraleSupélec

last update: August 25, 2026

Contents

1 Artificial Intelligence

3MD1540	Machine learning 1	35.5 h	3 ECTS
3MD4010	Machine learning 2	23.0 h	2 ECTS
3MD4040	Deep learning	28.5 h	3 ECTS
3MD1530	Statistical Models 1	29.0 h	3 ECTS
3MD4050	Statistical models 2	33.0 h	3 ECTS

2 Computer Science

SPM-INF-003	Introduction to C/C++ Programming	41.5 h	4 ECTS
SPM-INF-013	Operations research	19.5 h	2 ECTS
SPM-INF-014	Langage processing	21.0 h	2 ECTS

3 Projects

SPM-PRJ-014	Scientific Project	170.0 h	15 ECTS

Machine learning 1

Course supervisor: Hervé Frezza-Buet

Total: 35.5 h , 3 ECTS **CM:** 18.5 h, **TP:** 15.0 h

3MD1540 *back*

Description: This course sets out the general framework of machine learning, allowing you to situate the different approaches in the field. It covers the notions of data pre-processing, an introduction to statistical learning theory (risks, overlearning, convex proxies, regularization), the difference between frequentist and Bayesian approaches, supervised, unsupervised, semi-supervised and reinforcement learning paradigms. Some approaches are detailed (Kernel methods, SVM, Boosting, Bagging, Decision trees...).

Learning outcomes: At the end of this course, students will be able to recognize the different classes of algorithms in the landscape of the many methods available on the shelf. They will also have the statistical notions that will enable them to make reasoned use of these methods, thus avoiding a black-box approach with blind parameter testing.

Means: The courses and practical work are given by Hervé Frezza-Buet, Arthur Hoarau, Jérémy Fix. The courses present theoretical aspects, mathematical proofs, but are also illustrated by demonstrations of algorithms. The practical work will be done in Python, using sickit-learn, in pairs.

Evaluation methods: 2h written test, can be retaken.

CM:

1. Datasets and learning (1.5 h)
2. Frequentist, Bayesian, evaluation (1.5 h)
3. Risks (1.5 h)
4. C-SVC, Lagrange formulation (1.5 h)
5. Kernels, numerical resolution (1.5 h)
6. SVMs for regression, unsupervized learning, nu-versions of SVMs. (1.5 h)
7. Intro supervisé (1.5 h)
8. Arbres de décision (1.5 h)
9. Réduction de dim. (1.5 h)
10. Partitionnement/Regroupement (1.5 h)
11. Quantification vectorielle (1.5 h)
12. Intervention (2.0 h)

TP:

1. Data Science en Python (3.0 h)
2. Arbres de décision (3.0 h)
3. Non Supervisé (3.0 h)
4. TP+ 1/2 (3.0 h)
5. TP+ 2/2 (3.0 h)

Machine learning 2

Course supervisor: Arthur Hoarau

Total: 23.0 h , 2 ECTS **CM:** 9.0 h, **TP:** 12.0 h

3MD4010 *back*

Description: This course complements Machine Learning 1 with notions of data processing (dimension reduction, etc.), unsupervised learning, active and semi-supervised learning, explicability issues.

Learning outcomes: By the end of this course, students will have completed their breadth approach to machine learning.

Evaluation methods: 2h written test, can be retaken.

CM:

1. Bagging (1.5 h)
2. Boosting (1.5 h)
3. Détection anomalies (1.5 h)
4. Quantification Incertitude (1.5 h)
5. Apprentissage semi-supervisé (1.5 h)
6. Explicabilité (1.5 h)

TP:

1. Bagging (3.0 h)
2. Forêts aléatoires (3.0 h)
3. Détection anomalies/OOD (3.0 h)
4. semi-supervisé/XAI (3.0 h)

Deep learning

Course supervisor: Jérémy Fix

Total: 28.5 h , 3 ECTS **CM:** 13.5 h, **TP:** 15.0 h

3MD4040 *back*

Description: Deep learning is a technology that is booming, thanks in particular to the use of GPUs (Graphical Processing Units), the availability of large amounts of data and the understanding of theoretical elements that make it possible to better define neural network architectures that are more easily trainable. In this course, students will be introduced to the basics of neural networks and also to the different architectural elements that make it possible to design a neural network according to the prediction problem considered. The course is divided into modules in which questions of optimization algorithms, their initialization, regularization techniques, fully connected architectures, convolutional networks, recurrent networks, introspection techniques are addressed. Practical works on GPUs are associated with the courses.

Content: The lectures will be discussing :

- Historical introduction to neural networks, linear classifier/regressor (1.5 HPE)
- Computational graph and gradient descent, Fully connected networks, RBFs, Auto-encoders, Optimisation methods, initialisation, regularisation (3 HPE)
- Convolutional networks: architectures (1.5 HPE)
- Convolutional networks: classification, object detection, semantic segmentation (1.5 HPE)
- Recurrent networks: architectures and training (1.5 HPE)
- Recurrent networks: applications (1.5 HPE)
- Generative models: autoregressive models, VAE, GANs and diffusion (1.5 HPE)

The praticals will be on:

- Introduction to pytorch on classification with linear predictors, fully connected networkjs and convolutional networks (3 HPE)
- Convolutional neural networks for semantic segmentation (3 HPE)
- Recurrent neural networks applied to sequence to sequence translation (3 HPE)
- Adversial neural networks (3 HPE)
- Neural implicit representations (3 HPE)

Learning outcomes: Being able to implement and deploy a deep learning algorithm.
Being able to choose the right architecture that suits a particular machine learning problem.
Being able to diagnose the training of a neural network (what is it learning ? how is it learning ? is it learning ? will it be able to generalize ?)

Teaching methods: The course is structured into lectures during which we introduce the theoretical and experimental notions are introduced and illustrated with various examples. Practicals allow the students to put into practice the notions we discuss during the lectures.

Means: We will be using the Pytorch framework. The students could work in pairs and will make use of GPUs of the Data Center d"Enseignement of the Metz campus for running their codes.
A page will be dedicated on edunao. Forums will be opened, allowing the students to ask questions on the lectures or the tutorials, having the possibility to interact with the teaching staff but between them as well.

Evaluation methods: The assessment will be based on two elements: assessments on paper questionnaires at the beginning of each practical session (1/3 of the total grade) and participation to a dedicated challenge in a team (2/3 of the total grade). The grade depends both on their submission, and a recorded video explaining their approach and results.

External resources:

- Site du cours
- Sujets des TPs

CM:

1. Introduction et réseaux linéaires (1.5 h)
2. Graphe de calcul, descente de gradients et réseaux feedforward (1.5 h)
3. Optimisation, Initialisation, Régularisation (1.5 h)
4. Eléments d'architectures des réseaux convolutifs (1.5 h)
5. Applications des réseaux convolutifs à la détection d'objets et la segmentation sémantique (1.5 h)
6. Les approches par transformers (1.5 h)
7. Eléments d'architecture des réseaux récurrents (1.5 h)
8. Architectures et applications des réseaux récurrents (1.5 h)
9. Modèles génératifs : modèles auto-regressifs, VAE, GANs et diffusion (1.5 h)

TP:

1. Introduction à PyTorch par la classification d'images (3.0 h)
2. Réseaux convolutifs pour la segmentation sémantique (3.0 h)
3. Application des RNNs pour la retranscription de la parole (3.0 h)
4. Modèles génératifs (3.0 h)
5. Représentations Neurales Implicites (3.0 h)

Statistical Models 1

Course supervisor: Frédéric Pennerath

Total: 29.0 h , 3 ECTS **CM:** 16.5 h, **TD:** 4.5 h, **TP:** 6.0 h

3MD1530 *back*

Description: The ”Statistical Modeling” courses ModStat1 and ModStat2 deal with the modeling of systems for which the outputs are sufficiently uncertain that they need to be modeled by random variables. The course begins with a review of statistics and the introduction of elementary models (e.g. naive Bayes, linear regression, etc.), moving progressively towards more complex models. While the courses present the most useful elementary models and methods in this modeling context, they are not intended to be an exhaustive catalog. The aim is rather to present, within a consistent theory, the concepts and tools common to all these models and methods, and to show how, starting from modeling hypotheses specific to each concrete problem, these concepts are logically assembled before leading to an operational method. From a practical point of view, the aim of this course is not only to give students the means to understand and make good use of existing model implementations, but also to design their own implementations to take into account the specificities of a given problem. The course focuses on linking theory to practice: first, the hypotheses associated with a given class of problems are identified in class, followed by theoretical modeling work, leading to the definition of a model and its estimation algorithms. These results are then applied to a case study in tutorial sessions, before being implemented (in Python) and evaluated on data in practical exercises. The ModStat1 course will introduce the basic tools of statistical modeling, while the ModStat2 course will focus on hidden variable models.

Prerequisites: - Basic knowledge of probability theory, statistics and machine learning - Beginner level in Python / Numpy programming

Learning outcomes: - Be able to choose a statistical model/method adapted to the problem under consideration and implement it appropriately - Be able to understand the theoretical concepts underlying a statistical inference method presented in a scientific article. - Be able to implement a model / statistical method in a language such as Python. - Be able to adapt a model/method to take into account the specificities of the problem being addressed.

Evaluation methods: 2h written test without documents, can be retaken.

CM:

1. Modèles statistiques (1.5 h)
2. Estimation (1.5 h)
3. Estimation bayésienne (1.5 h)
4. Réseaux bayésiens 1 (1.5 h)
5. Naive Bayes (1.5 h)
6. Réseaux bayésiens 2 (1.5 h)
7. Causalité (1.5 h)
8. Modèles gaussiens (1.5 h)
9. Modèles linéaires (1.5 h)
10. Famille exponentielle (1.5 h)
11. GLM (1.5 h)

TD:

1. Estimation bayésienne (1.5 h)
2. Modélisation causale (1.5 h)
3. Régression (1.5 h)

TP:

1. Modélisation et modèles gaussiens (3.0 h)
2. Régression (3.0 h)

Statistical models 2

Course supervisor: Frédéric Pennerath

Total: 33.0 h , 3 ECTS **CM:** 12.0 h, **TD:** 6.0 h, **TP:** 12.0 h

3MD4050 *back*

Description: This course is an extension of the ModStat1 course. It is structured around the three fundamental concepts of stochastic processes, latent variables and approximate inference techniques. The first part of the course on processes focuses on three main families of processes: point processes, Markov processes and Gaussian processes. The notion of latent variable is then addressed through mixture models and the EM algorithm. The two notions are then combined to develop hidden Markov models, for both discrete (HMM) and continuous states (Kalman filters). Finally, approximate inference techniques are presented, with sampling techniques (MCMC) and variational inference.

Prerequisites: - Having followed the course "Statistical Models 1" - Beginner level in Python / Numpy programming

Learning outcomes: At the end of this course, students will be able to associate the corresponding type of stochastic process with data series and apply the associated estimation methods. They will also be able to specify a model incorporating hidden variables and apply the EM algorithm to estimate its parameters. They will be able to model a clustering problem in the form of a mixture model. They will be able to specify an HMM or a Kalman filter to model the dynamic behavior of a discrete or continuous state system.

Evaluation methods: 3h written test with documents, can be retaken.

CM:

1. Processus ponctuels (1.5 h)
2. Processus de Markov (1.5 h)
3. Processus gaussiens (1.5 h)
4. Modèles de mélanges (1.5 h)
5. Modèles de Markov cachés (1.5 h)
6. Filtre de Kalman (1.5 h)
7. Echantillonnage (1.5 h)
8. Inférence variationnelle (1.5 h)

TD:

1. Processus de Poisson et de Markov (1.5 h)
2. Modèles de Markov cachés (1.5 h)
3. Filtre de Kalman (1.5 h)
4. Echantillonnage (1.5 h)

TP:

1. Processus gaussiens (3.0 h)
2. Modèles de mélange (3.0 h)
3. Filtre de Kalman et particul. (3.0 h)
4. Echantillonnage (3.0 h)

Introduction to C/C++ Programming

Course supervisor: Hervé Frezza-Buet

Total: 41.5 h , 4 ECTS | **CM:** 10.5 h, **TD:** 3.0 h, **TP:** 28.0 h

SPM-INF-003 *back*

Description: This course covers C programming, focusing on execution threading (loops, function calls, recursive functions), memory manipulation (structured types, pointers, stack and heap, binary representations). The first steps towards object-oriented design (encapsulation without the syntax of an object language like C++). This course also covers the topic of separate compilation (headers, external variables, linkage, dynamic libraries, etc.).

Learning outcomes: On completion of this course, students will be able to write, compile and debug C/C++ programs involving the basic elements of the language.

Evaluation methods: Assessment based on participation in experiments and results.

External resources:
- C++ web pages

CM:

1. Fil d'exécution, pile 1/2 (1.5 h)
2. Fil d'exécution, pile 2/2 (1.5 h)
3. Organiser la mémoire 1/2 (1.5 h)
4. Organiser la mémoire 2/2 (1.5 h)
5. Compilation séparée (1.5 h)
6. Définition de types et encapsulation (1.5 h)
7. STL et smart pointers (1.5 h)

TD:

1. Collections 1/2 (1.5 h)
2. Collections 2/2 (1.5 h)

TP:

1. Fil d'exécution, pile (4.0 h)
2. Ranges (4.0 h)
3. Simulateur de réseaux de Petri (4.0 h)
4. Rendu graphique de height fields (4.0 h)
5. Rendu graphique de height fields (4.0 h)
6. Finitions des TPs (4.0 h)
7. Finitions des TPs (4.0 h)

Operations research

Course supervisor: Christophe Rapine

Total: 19.5 h , 2 ECTS **CM:** 6.0 h, **TP:** 12.0 h

SPM-INF-013 *back*

Description: The course is an introduction to combinatorial optimization problems in the applied context of operations research. The latter covers formal optimization methods useful to organizations (companies, etc.) for making rational decisions, whether in logistics, strategy (investment plans, etc.), economics, etc. This course and the optimization course are inseparable in the sense that they divide up the presentation of the various optimization techniques between them, depending on the nature of the problems these techniques solve: while the optimization course focuses on numerical analysis methods that optimize a continuous function defined in a Euclidean space, RECHOP addresses combinatorial optimization problems in a discrete space, the solution of which is more algorithmic. As a result, this course places a strong emphasis on algorithms and their complexity and can therefore be seen as an extension of the first-year course "Data Structures and Algorithms" (DSA). The course will begin by addressing combinatorial problems within the very general framework of constraint programming. We will then focus on instances of problems for which a polynomial-time solution exists, such as dynamic programming or graph matching problems. We will also develop efficient approximation algorithms for the difficult problem of integer linear programming after linear programming has been covered in the optimization course, thus linking the two courses.

Learning outcomes: At the end of this course, students will have a general understanding of the types of problems encountered in operational research. They will be able to formalize an operational research requirement and recognize the precise nature of the underlying combinatorial problem. Where appropriate, they will be able to match this combinatorial problem with a suitable solution technique.

Evaluation methods: 1h30 written test, can be retaken

CM:

1. Programmation par contraintes (1.5 h)
2. Problèmes d'optimisation temporelle. Programmation dynamique (1.5 h)
3. Problème d'affectations: couplage de graphes et mariages stables (1.5 h)
4. Programmation linéaire entière (1.5 h)

TP:

1. Programmation par contraintes (3.0 h)
2. Programmation dynamique (3.0 h)
3. Problèmes de couplage (3.0 h)
4. Programmation linéaire entière (3.0 h)

Langage processing

Course supervisor: Benoît Valiron

Total: 21.0 h , 2 ECTS **CM:** 7.5 h, **TD:** 4.5 h, **TP:** 9.0 h

SPM-INF-014 *back*

Description: The objective is to introduce the concepts of language theory and put them into practice so that students fully understand and see the value of these concepts in dealing with concrete problems. We will cover regular languages, grammars, and how a compiler works.

Learning outcomes: By the end of this course, students will have understood the relevance of language theory in computer science.

Evaluation methods: Assessment based on participation in experiments (TD/TP) and results obtained

---

CM:

1. Expressions régulières, langage reconnu par un automate 1/2 (1.5 h)
2. Expressions régulières, langage reconnu par un automate 2/2 (1.5 h)
3. Grammaires (1.5 h)
4. Descente récursive, parseur LL (1.5 h)
5. Fonctionnement d'un compilateur (1.5 h)

TD:

1. Grammaires (1.5 h)
2. Descente récursive, parseur LL (1.5 h)
3. Compilation (1.5 h)

TP:

1. Reconnaissance de tokens dans un document (3.0 h)
2. Evaluation d'expressions arithmétiques (3.0 h)
3. Mise en oeuvre du compilateur sur machine (3.0 h)

Scientific Project

Course supervisor: Hervé Frezza-Buet, Damien Rontani

Total: 170.0 h , 15 ECTS **Projet:** 170.0 h

SPM-PRJ-014 *back*

Description: The objective of this project is to carry out ambitious scientific and/or technological work, which may consist of either an exploratory study or the development of advanced technical solutions.

Learning outcomes: By the end of this project, students will have gained an understanding of the methodology for approaching a scientific topic (state of the art, identification of appropriate tools, planning of the work to be done, analysis of experiments, and communication of results).

Evaluation methods: Assessment based on the deliverables, the report, and the defense.