ESIROI
ÉCOLE D'INGÉNIEURS
UNIVERSITÉ DE LA RÉUNION

DEPARTMENT : ***IT***
SEMESTER : ***S8***

IDENTIFICATION

Course code : ESI-SPI-CI-IN4-S8-UE5-EC1
ECTS : 2

DURATION

Lectures :	4
Tutorial classes :	18
Labs :	0
Total :	22

Project :
Personnal work :

ASSESSMENT

Continuous assessments and evaluated practical work.

TEACHING MATERIALS

Slides, guided exercises (*travaux dirigés*), and practical work (*travaux pratiques*) handouts.

LANGUAGE

English

CONTACT

Mathilde LEPETIT
mathilde.lepetit@univ-reunion.fr

Modifié le : 14 juillet 2026

ESIROI - Campus de Terre Sainte
40 Avenue de Soweto, 97455 Saint-Pierre, Cedex
Tel.: +262 262 81 83 01 / 02 ; www.esiroi-reunion.fr

UE : UE8-DATA

COURSE : Deep Learning

SCOPE OF THE COURSE

OBJECTIVES

Master the architectures and techniques of deep learning to design, train, and deploy neural network models suited to complex data science problems. This course deepens the theoretical foundations of multilayer networks and covers major architectures — convolutional, recurrent, and generative — with a focus on practical challenges specific to large-scale learning : regularization, optimization, interpretability, and scaling on GPU and distributed systems.

COMPETENCE AND SKILLS

- Design and implement deep neural network architectures (feedforward, CNN, RNN, LSTM) suited to the nature and volume of data for a given problem
- Train, regularize, and optimize a deep learning model by applying best practices : initialization, *dropout*, *early stopping*, parameter sharing, hyperparameter tuning
- Define, implement, and evaluate a predictive analysis leveraging massive data using deep learning frameworks executed on GPU or distributed systems
- Apply convolutional (CNN) and recurrent (RNN/LSTM) architectures to multimedia data analysis : image classification, object detection, signal and time series processing
- Select appropriate deep learning tools and frameworks (TensorFlow, PyTorch, Keras) considering constraints of data volume, performance, and interpretability
- Stay informed about the state of the art in deep learning and assess the relevance of advanced architectures (generative models, transformers) for new use cases

PROGRAM

- **Foundations of deep neural networks :**
  - Review of the multilayer perceptron (MLP) ; hierarchical representation of features.
  - Activation functions : ReLU, sigmoid, tanh, softmax — properties and use cases.
  - Backpropagation algorithm in a deep *feedforward* network.
  - Gradient problems : vanishing and exploding gradients ; solutions (batch normalization, weight initialization).
  - Regularization : *dropout*, *early stopping*, *weight decay*, parameter sharing.
- **Convolutional neural networks (CNN) :**
  - Convolution principle : local pattern detection (edges, textures) ; translation invariance.
  - Convolution layers, *pooling* layers (max pooling, average pooling), and flattening (*flatten*).

- Reference architectures : LeNet, AlexNet, VGG, ResNet, EfficientNet.
- Transfer learning and *fine-tuning* on limited data.
- Applications : image classification, object detection, semantic segmentation.

— **Recurrent neural networks (RNN / LSTM / GRU) :**

- RNN architecture ; backpropagation through time (BPTT).
- Challenge of long-term dependencies ; vanishing gradient problem in RNNs.
- LSTM (*Long Short-Term Memory*) and GRU (*Gated Recurrent Unit*) : gating mechanisms and use cases.
- Applications : time series processing, signals, sequence generation.

— **Optimization and scaling :**

- Optimization algorithms : SGD, Momentum, RMSprop, Adam.
- Learning rate scheduling.
- GPU execution : CUDA, tensor computation parallelization.
- Scaling on distributed systems : data and model parallelism (link with DAT701).
- Tool selection based on data volume : TensorFlow, PyTorch, Keras, JAX.

— **Interpretability and best practices :**

- Visualization of activations and convolutional filters ; saliency maps (*Grad-CAM*).
- Model evaluation : metrics, learning curves, overfitting detection.
- Deployment of a deep learning model : export, quantization, optimized inference.

— **Advanced topics (subject to availability) :**

- Generative models : generative adversarial networks (GAN) — architecture, training, applications (image synthesis, data augmentation) ; practical challenges (*mode collapse*, convergence instability).
- Introduction to Transformers and attention mechanisms ; vision transformers (ViT) and language models (LLM).
- Self-supervised and contrastive learning ; *foundation models*.

BIBLIOGRAPHY

- Kubat, M. (1999). Neural networks : a comprehensive foundation by Simon Haykin, Macmillan, 1994, ISBN 0-02-352781-7. The Knowledge Engineering Review, 13(4), 409-412.
- Michael A. Nielsen, “Neural Networks and Deep Learning”, Determination Press, 2015
- Simon Haykin - *Neural Networks and Learning Machines* - Prentice Hall, 3rd ed., 2008
- Michael A. Nielsen - *Neural Networks and Deep Learning* - Determination Press, 2015 : `http://neuralnetworksanddeeplearning.com`
- Ian Goodfellow, Yoshua Bengio & Aaron Courville - *Deep Learning* - MIT Press, 2016 : `https://www.deeplearningbook.org`

— Aurélien Géron - *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow* - O'Reilly, 3rd ed., 2022
— Kubat, M. - *An Introduction to Machine Learning* - Springer, 3rd ed., 2021
— PyTorch documentation : `https://pytorch.org/docs/stable/`
— TensorFlow / Keras documentation : `https://www.tensorflow.org/api_docs`
— Papers With Code — State of the art in deep learning : `https://paperswithcode.com`
— Distill.pub — Interactive deep learning visualizations : `https://distill.pub`
— Fast.ai — Practical deep learning courses : `https://www.fast.ai`

REQUIREMENTS

— **Machine Learning (S7) :** classification and regression algorithms, model evaluation, bias/variance trade-off, cross-validation — direct and essential foundation for this course.
— **Massive Data Systems (S7) :** distributed architectures, parallel computing frameworks — necessary for scaling deep learning models.
— **Data Mining (S6) :** data preprocessing, feature extraction, introduction to clustering.
— **Probability and Random Variables (S5) :** probability distributions, expectation, variance — foundations of loss functions and optimization methods.
— **Stochastic Processes (S6) :** sequence and time series modeling — useful for understanding RNNs and LSTMs.
— **Expected cross-disciplinary skills :**

  — Advanced Python programming (numpy, pandas, matplotlib, scikit-learn)
  — Linear algebra : matrix products, decompositions, eigenvalues
  — Differential calculus : partial derivatives, chain rule, gradient
  — Basic knowledge of GPUs and computing environments (Google Colab, Jupyter)
  — Ability to read technical documentation and research papers in English