Module Code	EE5C16/EEU44C16
Module Name	Deep Learning and Its Applications
ECTS Weighting[1]	10 ECTS
Semester taught	Semester 1
Module Coordinator/s	Assistant Prof. François Pitié
Module Learning Outcomes with reference to the Graduate Attributes and how they are developed in discipline	On successful completion of this module, students should be able to: 1. Describe the main neural network architectures used in applications today. 2. Describe the parameters used in popular Deep Learning software libraries such as Keras. 3. Implement neural network applications using python 3 and Keras. 4. Evaluate the performance of Machine Learning algorithms and analyse the potential pit falls. 5. Design neural network architectures for a particular application. 6. Train and debug neural network models (e.g. detect overfitting and provide a solution to mitigate its effects). 7. Plan and design solutions for industry projects that require neural net technology. 8. Answer typical questions from job interviews on Deep Learning. 9. Analyse the ethical, social, and environmental implications of deploying Deep Learning systems in real-world engineering applications, and demonstrate responsible decision-making in model design and data usage. 10. Apply basic engineering management practices such as version control, and resource planning to manage Deep Learning projects effectively and professionally. **Graduate Attributes: levels of attainment** To act responsibly - Attained To think independently - Attained To develop continuously - Attained To communicate effectively - Attained

[1] TEP Glossary

Module Content	This module is an introduction to Machine Learning with a strong focus on Deep Learning. Deep learning is another name for machine learning using artificial neural networks. It is loosely inspired by the structure of the neurons in the cerebral cortex. Deep Learning has recently become a disruptive technology that has been taking over operations of technology companies around the world and is having a profound impact all aspects of society. When you read or hear about AI or machine Learning in the news, it really refers to Deep Learning technology. Learning how to master Deep Learning is increasingly becoming as important as learning how to code. The course starts with an introduction to some essential aspects of Machine Learning, including Least Squares, Logistic Regression and a quick overview of some popular classification techniques such as decision trees and SVMs. The course then quickly dives into the fundamentals of Neural Nets, including Feed Forward Neural Nets, Convolution Neural Nets, Recurrent Neural Nets and Transformers. Students get introduced to the most modern architectures in the later part of the module, Large Language Models. The material has been constructed in collaboration with leading industrial practitioners including Google, YouTube and Intel, and students typically get guest lectures from practitioners and ethicists. Hands on labs will give you experience with the field and allow you to develop your own Deep Learning applications.
Teaching and Learning Methods	*Teaching Strategy* Deep Learning has become so successful partly because it does not require arcane knowledge to practice it. This course aims at popularising Deep Learning mastery to a class of students with diverse technical background (e.g. to both computer science and engineering students). The teaching strategy for this module is a mixture of lectures, problem-solving tutorials and laboratories dedicated to implement and solve machine learning problems. Most of the theoretical elements of Machine Learnings and Deep Learning will be covered in the first half of the term. The rest of the term is dedicated to expose the students to more advanced labs and industry related problems. The students will have access to online resources and recorded lecture videos. The teaching strategy includes elements of Mastery Learning, mixed with Flipped Classroom. Labs are automatically assessed, and students are allowed unlimited submissions. Students must also successfully pass several MCQs that checks the mastery of the fundamentals.

Lab Environment

We have developed specifically for this module a web application, so that students can learn best industry practices:

- Programming is done in python 3 using Keras and TensorFlow. Everything is running on Colab, which gives on-demand scalable computing resources.
- The coding environment is a combination of shell/terminal, editor, and Jupyter notebook.
- Git is used to checkpoint lab progress and give continuous feed-back on lab assignments.

The platform smooths out all the painful installation and configuration parts so that students can focus on the essential.

Assessment Details[2]
Please include the following:

- **Assessment Component**
- **Assessment description**
- **Learning Outcome(s) addressed**
- **% of total**
- **Assessment due date**

Assessment Component	Assessment Description	LO Addressed	% of total	Week due
Exam	End of Year Exam, 2-hr in-person	LO1-8	60%	End of Year
CA	Lab Submissions & Fundamental Knowledge Tests	LO1-6,9,10	20%	Spaced over Semester
CA	Mid-Term Exam	L01-8	20%	Week 8

Reassessment Requirements

100% based on exam

Contact Hours and Indicative Student Workload[2]

Contact hours: 66
Independent Study (preparation for course and review of materials): 80
Independent Study (preparation for assessment, incl. completion of assessment): 120

Recommended Reading List

Textbooks

Supporting references (research publications, press articles, YouTube videos) are included in the handouts.

- Deep Learning, Ian Goodfellow et al., (MIT press), [https://www.deeplearningbook.org]

[2] TEP Guidelines on Workload and Assessment

	• Machine Learning on Cousera, Andrew Ng [https://www.coursera.org/learn/machine-learning] • Neural Networks and Deep Learning, Michael Nielsen [http://neuralnetworksanddeeplearning.com/] *Website and Video of Lectures* • Lecture Notes are available at https://frcs.github.io/4C16-LectureNotes/ · • Older Recorded videos are available on this YT playlist: https://goo.gl/DP2jnJ. Lectures will be recorded and available on Teams.
Module Pre-requisite	
Module Co-requisite	
Module Website	https://frcs.github.io/4C16-LectureNotes/
Are other Schools/Departments involved in the delivery of this module? If yes, please provide details.	
Module Approval Date	
Approved by	Prof. Naomi Harte
Academic Start Year	September 2025
Academic Year of Date	2025/2026