Course Title: Deep Learning

Course number: ECE 510

Course credits: 3

Instructor: Manel Martínez-Ramón **Email**: manel@unm.edu

Office Location: ECE237b, Department of Electrical and Computer Engineering.

Office Hours: MON-WED 11 AM to 12 PM

Office phone: (505) 277-3008

Term / Semester: Spring

Course Description and Contents:

This course is intended to be an introduction to deep learning, where the student will learn the theoretical fundamentals of this discipline and will acquire the basic skills necessary to start developing their own deep learning designs in Python. The course will include programming in Pytorch and Tensorflow with which the student can become a serious deep learning developer.

The grading of the course will be based on the projects assigned to the student. The grading will take into consideration the quality of the experimental results, as well as the ability of the student to present them in a compelling and professional way in written documents. Extensive guidance will be provided on techniques for writing scientific and technical documents.

Module 1. Introduction to deep learning

This module introduces the basic multilayer perceptron that originated the development of deep learning. The neuron activation of hidden nodes and output and their implications are presented. The maximum likelihood learning criterion and backpropagation algorithm are fully developed. Then, basic additional concepts of deep learning are introduced, such as machine complexity and regularization for deep learning.

1.1 The perceptron

Example 1.1 : Perceptron rule

1.2 Structure and optimization criteria

1.3 The backpropagation algorithm

Example 1.2 A simple nultilayer perceptron

1.4 Training practicalities

Example 1.3. Common optimizers

Module 2. Deep learning tools

Introduction to Tensorflow, Keras, and Pytorch. Here the concepts and orientation of the different options to develop deep learning techniques are introduced from the point of view of development speed, efficiency, and parallel processing power. This module assumes that the student has reviewed Module 1, which implies that they have been introduced to the concepts of structure, criteria, and algorithms. Also, students have had the opportunity to see some basic Python codes containing at least a class with methods and an instantiation of it to be used in the examples and exercises, without needing to understand their Python structure. In this module, we introduce the basic elements of Python to be used throughout the course, and we will revisit the code previously introduced in the previous module, among other examples.

2.1 Python: an overview

Example 2.1. Basic Python

2.2 NumPy

Example 2.2. A NN in NumPy

2.3 MatplotLib

2.4 SciPy

2.5 Scikit-learn

Example 2.5.1 Feature selection with Scikit-learn

Example 2.5.2 ML in Scikit-learn (1)

Example 2.5.3 ML in Scikit-learn (2)

2.6 Pandas

2.7 Seaborn

2.8 Tensorflow and Keras

Example 2.8.1 Regression in Tensorflow and Keras

Example 2.8.2 A neural network workflow in Keras

2.9 Pytorch

Example 2.9 A NN in Pytorch

Module 3. Convolutional Neural Networks

In this module, the convolutional neural network is introduced in a parsimonious way, by first developing the convolution operation as it is used in deep learning and its introduction in the structure of a CNN. After, the pooling concept is added to the CNN process, and convolution and pooling are explained from a probabilistic point of view. Convolution algorithms and practical basic CNN implementations constitute the experimental part of this module. In this module, we also review the most popular CNN structures, such as Resnet/AlexNet, VGG16, and others. The experiments include the practice with one of these structures.

3.1 Elements of the convolutional neural network

3.2 Operational elements of the CNN

Example 3.2.1. An image convolution

Example 3.2.2. An image convolution with the torch package

Example 3.2.3. A CNN for classification

3.3. Advanced CNN structures

Module 4. Sequence modeling with recurrent neural networks

Here we cover the basics of Recurrent neural networks. The module starts with the architecture of the Recurrent Neural network and then explains how these networks are used for modeling sequential information. Further, into the module, the training criterion is introduced which describes the feed-forward training, loss functions, and backpropagation through time. Next, the different types of RNNs and their applications are discussed. The following section explains the shortcomings of RNNs and highlights the details of different types of gradient problems and the solutions to these problems. The next section discusses the details of other RNN-derived structures that were introduced to mitigate the short-term memory problem associated with traditional RNNs.

4.1 Structure of the Elman RNN

4.2a. Training an RNN. Recursive gradients.

4.2b Training an RNN. The backpropagation through time.

Example 4.2.1 RNN using Pytorch

4.3. Deep RNN and Bidirectional RNN

Example 4.2.2. Deep RNN using Pytorch

Example 4.2.3 Bidirectinal RNN using Keras

4.4. Long Short-Term Memory Networks

Example 4.2.4 LSTM using Keras and Pytorch

4.5. Gated Recurrent Units

Example 4.2.5 Sentiment analysis with GRU

4.6 Machine translation with recurrent neural networks

Module 5. Attention-based networks

This module provides a structured and comprehensive overview of the developments in attention-based networks. The first section summarizes the different types of attention mechanisms based on sequence, levels, positions, and representations. Finally, we review the network architectures that widely use attention and also discuss a few applications in which attention-based networks have shown a significant impact.

5.1. Attention mechanisms

5.2. Transformers

5.3. Transformers for vision

5.4. Large-scale pretraining with transformers

Module 6. Generative Adversarial Networks

This module covers generative adversarial networks. Primarily, it introduces the two elements of GANs namely discriminator and generator. The complete architecture of the GAN is illustrated in the following section to have a higher level of understanding of the network. Next, the training criteria are outlined which describe the alternate training process between the discriminator and the generator. The loss functions that model the probability distribution of the data are also added in this section. Finally, popular models derived from GAN are presented and the chapter is concluded by summarizing the advantages and trade-offs of GAN. The following alternative options are proposed to avoid undesired overlappings with concurrent ML courses in the department

6.1 Introduction to GANNS

6.2 Examples of GANNS

Course goals:

This course is included in the ECE areas of Computer Vision and Image Processing (CVIP), and Information Systems (IS). In these areas, machine learning plays an important role since the techniques described in this discipline are state-of-the-w tools. The use of deep learning in CVIP is the most prominent approach, with unique performance never unseen before. Besides, the use of DL in IS is an already mature solution for information processing and knowledge extraction in any IS application.

Throughout the semester, students will be learning fundamentals of deep learning through theoretical introduction and practice. The student will not only learn the fully theoretical background of DL, but they will acquire skills necessary in DL research and applications, including practical training in CVIP and IS.

The course goals can thus be summarized as

Knowledge

The student will acquire the theoretical principles of DL
The student will study and know in detail the main DL structures and algorithms in use today.

Skills

The students will use the most well-known and common DL tools, based on Python programming
The students will practice with academic and real applications of DL with a variety of problems and databases.

Learning Objectives

C1. Ability to describe and explain the deep learning theory and techniques described in the course.

C2. Solve practical deep learning problems by the use of deep learning techniques.

C3. Ability to conduct experiments with deep learning, analyze and interpret the results, and report them in a technical document and orally.

Textbooks/Supplies/Materials/Equipment/Technology or Technical requirements

Textbook:

Manel Martinez-Ramon, Aswathy Rajendra Kurup, Meenu Ajith. Deep Learning, A Practical IntroductionLinks to an external site., Wiley, 2024 (recommended, not required)

Alternatively, students can use book Bishop & Bishop, Deep Learning. Foundations and Concepts

Supplementary Textbooks:
The instructor will be using materials from the following textbook. Students are not required to purchase them:

Eli Stevens and al., Deep Learning with PyTorch, Manning, 2020.
Ian Goodfellow et al. Deep Learning. MIT, 2016.
Aston Zhang, Zack C. Lipton, Mu Li, Alex J. Smola, Dive Into Deep LearningLinks to an external site.. This textbook is an open-source project and it can be found at http://d2l.ai/Links to an external site..

Additionally, the books Hands-On Machine Learning with Scikit-Learn & Tensorflow and Introduction to Machine Learning with Python are freely available to all UNM students. To access them, simply go to https://libguides.unm.edu/Safari
There are several other Python textbooks available through this link.
The instructor will recommend in the future the use of his book, to be published with Wiley. Conflicts of interest will be cleared by following UNM policies.

Technical requirements

A high-speed Internet connection is highly recommended.
Supported browsers include Internet Explorer, Firefox, and Safari. Detailed Supported Browsers and Operating Systems: http://online.unm.edu/help/learn/students
Any computer capable of running a recently updated web browser should be sufficient to access your online course. However, bear in mind that processor speed, amount of RAM and Internet connection speed can **greatly** affect performance.
Many locations offer free high-speed Internet access including UNM's Computer Pods.
For using the Kaltura Media Tools inside Learn, be sure you have downloaded and installed the latest version of JavaLinks to an external site., FlashLinks to an

external site., and Mozilla FirefoxLinks to an external site.. They may not come preloaded.

Microsoft Office products are available free for all UNM students (more information on the UNM IT Software Distribution and Downloads page: http://it.unm.edu/software/index.html)

PYTHON is required.

Course requirements

Students will work on assigned homework corresponding to all chapters of the course. To be delivered before or on the given deadlines. Homework must be programmed in Python.

Students need to know the fundamentals of algebra, calculus, and probability theory. It is recommended to have passed the class “Machine Learning”, but it is not required. It is mandatory to have basic knowledge of Python.

Assessment and grading policy

The class will be graded according to the grades resulting from the assignments. Grading will be according to the corresponding rubrics. Students are allowed to resubmit an unlimited number of times before the deadline. Only the last submission will be graded. Students will be given written feedback. Assessments overdue will not be graded.

Plagiarism policy: documents with any level of plagiarism will be returned without grading. Homework with over 50% plagiarism will be graded with a final grade of 0. Students who incur plagiarism more than once will be failed (See "Academic Integrity statement" below).

All homework must be turned in with the following elements:

Experimental description
Results in a clear way
Discussion of the results
A copy of all Python scripts in a separate document in PDF. Screenshots of the scripts are not acceptable.
Handwritten work is not acceptable.

Students choosing the grade/no grade option will be also required to deliver the homework. Audit students are welcome upon proper UNM registration.

Letter grading will follow the UNM regulations:

Points (%)	Grade	Points (%)	Grade
[98,100]	A+	[80,83)	B-
[93,98)	A	[78,80)	C+
[90,93)	A-	[70,78)	C
[88,90)	B+	[0,70)	F
[83,88)	B		

Course schedule for semester Spring 2024 (TBD):

See Section Schedule in Canvas

The beginning and end of classes, spring break, and holidays as published in the campus calendar:

https://hr.unm.edu/docs/hr/campus-calendar-2023-2024.pdf

Attendance Policy

Regular and punctual attendance is required. UNM Pathfinder policies apply, which in part means instructor drops based on non-attendance are possible. This policy applies regardless of the grading option you have chosen.

Attendance will be controlled. Students with unjustified attendance of less than 80% will be graded proportionally to their attendance rate. The UNM attendance policy can be found in the link: https://pathfinder.unm.edu/campus-policies/class-absences-and-student-attendance.html

Cell Phones and Technology

Cell phones and other electronic devices must be muted before the beginning of class. The use of cell phones or other devices to chat, browse the Internet, or for any other purpose during the class is not acceptable. If you must receive a call, your cell phone must be switched to vibration.

Accommodation Statement:

By University Policy 2310 and the Americans with Disabilities Act (ADA), academic accommodations may be made for any student who notifies the instructor of the need for an accommodation. You must take the initiative to bring such needs to the instructor's attention, as he/she is not legally permitted to inquire. Students who may require assistance in emergency evacuations should contact the instructor as to the most appropriate procedures to follow. Contact the Accessibility Resource Center at 277-3506 for additional information.

Title IX Statement

UAP 2720 and 2740. Our classroom and university should always be spaces of mutual respect, kindness, and support, without fear of discrimination, harassment, or violence. If you ever need assistance or have concerns about incidents that violate this principle, please access campus support resources. These include confidential services at LoboRESPECT Advocacy Center, the Women's Resource Center, and the LGBTQ Resource Center. The University of New Mexico prohibits discrimination on the basis of sex (including gender, sex stereotyping, gender expression, and gender identity). UNM faculty and graduate teaching assistants are considered "responsible employees." "Responsible employees" must communicate reports of sexual harassment, sexual misconduct and sexual violence to Compliance, Ethics and Equal Opportunity. For more information on the campus policy regarding sexual misconduct, reporting, and reporting for "responsible employees," please see UAP 2720 and UAP 2740.

Academic Integrity Statement

Each student is expected to maintain the highest standards of honesty and integrity in academic and professional matters. The University reserves the right to take disciplinary action, up to and including dismissal, against any student who is found guilty of academic dishonesty or otherwise fails to meet the standards. Any student judged to have engaged in academic dishonesty in coursework may receive a reduced or failing grade for the work in question and/or for the course.

Academic dishonesty includes, but is not limited to, dishonesty in quizzes, tests, or assignments; claiming credit for work not done or done by others; hindering the academic work of other students; misrepresenting academic or professional qualifications within or without the University; and nondisclosure or misrepresentation in filling out applications or other University records.