EEE 498/591: Machine Learning Basics with Application to FPGAs

Contact Information and Office Hours:

Instructor: OL Hartin, Ph.D.

Contact Info: olhartin@asu.edu, Office GWC 340

Required Texts: Machine Learning Refined J. Watt (Cambridge University press 2016) second edition

Course Catalog Description: Machine learning has become an important element in many areas of engineering. The goal of this course is to provide a good mathematical background and give experience with using key packages and methods in Machine Learning. FPGAs provide an efficient method to deploy machine leaning models.

Course Overview: The goal of this course is to provide students with a firm mathematical and practical background for analyzing data, choosing and implementing Machine Learning solutions for problems (math skills are simple derivatives and basic linear algebra). Students will learn to use Python or Matlab libraries and for shallow learning and PyTorch for a deep learning project. Confronted with potential machine learning projects in the future they will be able to make good solution choices and understand the results.

Methods will be described to deploy these models in digital logic. This may be done using FPGAs or ASICs. This deployment will be described generally using Hardware Description Language (HDL) specifically System Verilog. Applications to physical problems will be discussed.

Quizzes and Exams

Class exams will be conducted in person, crib sheets with necessary information will be provided in the exam.

Quizzes will be taken individually in Canvas without time or other restrictions to make sure you are keeping up with the material.

Course Outcomes:

- Understanding of basic Machine Learning development, mathematics and methodology
- Understand code developed for different methods
- Understand deployment of machine learning models in digital logic

Enrollment Requirements:

Prerequisites: EEE203, PHY131, MAT342/343 Co-requisite: EEE350

And some programming language background, Java, C, C++, Python, or MatLab

Python will be used for in class examples and a simple tutorial will be provided, however, students may turn in their work in Visual Studio (C++ or C#), Matlab, or other.

Verilog/System Verilog background may be helpful but not required. There will be class discussion of programming FPGAs with Verilog. The objective of the course will not be to teach Verilog (there are other classes for that), but it may be used in projects.

You will not be required to purchase and FPGA prototype boards, units are available if needed in your project.

Course Learning Outcomes:

They will understand the basic methods and be able to develop and implement simple Machine learning algorithms, use algorithms provided in common libraries, understand and be able use deep learning algorithms and libraries such as PyTorch which is one the leading deep learning frameworks. Then these models will be implemented in Verilog on FPGAs.

Assignments:

Each student will have an account on the Research Computing Cluster (Agave), which is GPU enabled, to do any project/simulation they can't/or don't want to do on their own computer.

Coding in Python, Matlab, Pytorch and Verilog will be introduced and general coding solutions provided.

Homework 20% (8 homeworks equally weighted)
Project 35% (1 projects with several deliverables)

Midterm Exam 20% (in class exam covering fundamentals)
Final Exam 25% (in class exam covering fundamentals)

EEE591 students will be responsible (by department guidelines) for an additional homework.

The project will be defined in Canvas and vary from semester to semester to keep the course challenging and interesting. The project is done in groups of four, and groups may choose based on their interest, skills, or research.

There is no requirement to deploy to an FPGA in projects or homeworks.

Grading Procedure: Grades reflect your effort/performance on assignments and adherence to deadlines.

Notify the instructor BEFORE an assignment is due if an urgent situation arises and you are unable to submit the assignment on time. There are no makeup tests, so notify the instructor in advance of any absences.

Students are encouraged to start homework and project assignments as soon as available. Most students report that they should have stared a week or more earlier than they did.

Homework

There will be 8 homework sets, see Canvas Calendar for deadlines. They will be handed in online through Canvas. Each homework may contain programming and mathematical problems. Python will be most commonly used in class but simple parallels to Matlab and Visual Studio (also commonly used for Machine Learning) will be made where possible. Solutions will be provided after the due date.

Student homework assignment submissions made after the solution has been published **may** be accepted at a significant penalty. Though students that miss an assignment are encouraged to turn in a late submission.

Homework is graded with emphasis on effort.

Project

The project will be defined in Canvas and vary from semester to semester to keep the course challenging and interesting. The project is done in groups of four, and groups may choose based on their interest, skills, or research.

Grading Policy:

Your grade will be determined based on percentages of each assignment. Depending on the class performance and difficulty of the homework, projects and exams, I may curve the grade up, but will never curve down.

Required Primary and Secondary Materials:

Required Texts: Machine Learning Refined J. Watt (Cambridge University press 2016) second edition

Some specific additional topics from Python Machine Learning (S. Raschka Packt publishing 2017)

Good General Math Reference : CRC Standard Math Tables and Formulae 30th Ed, available pdf (http://pws.npru.ac.th/sartthong/data/files/CRC_Standard_Mathematical_Tables_and_Formulae__Thirtieth_Edition.pdf (Links to an external site.))

Required Software:

The course will largely use the open data science platform **Anaconda** powered by **Python**

Anaconda for Python, **or** Matlab for homeworks, for the project PyTorch **may** be used

Anaconda is available from:

https://www.anaconda.com/products/individual (Links to an external site.)

Matlab is available in MyApps

PyTorch is available at https://pytorch.org/ (Links to an external site.)for those running Anaconda

Modelsim and Quartus for application to FPGA, Verilog resources will be provided, FPGAs are provided in the lab (GWC 379) but the focus will be on software

Modelsim and Quartus are very useful int he later half of the course and may be downloaded at

https://fpgasoftware.intel.com/?edition=lite (Links to an external site.)

you will have to set up an intel account to download it

Other Good References Not Required:

Deep Learning, Goodfellow et al., - covers the mathematics of deep learning, available free online at http://www.deeplearningbook.org/. (Links to an external site.) (Links to an external site.)

And PyTorch.org