# MEEG 54403 Machine Learning for Mechanical Engineers

Fall 2024

Instructor: Han Hu; MEEG 106; Email: hanhu@uark.edu; Phone: 479-575-6790

Lecture Hours and Location: Mon/Wed/Fri 10:45 – 11:35 AM

Office Hours and Location: Mon/Wed 11:35 – 12:35 PM

**Course Description:** This course covers an introduction to supervised and unsupervised learning algorithms for engineering applications, such as visualization-based physical quantity predictions, dynamic signal classification, and prediction, data-driven control of dynamical systems, surrogate modeling, and dimensionality reduction, among others. The lectures cover the fundamental concepts and examples of developing machine learning models using Python and MATLAB. This course includes four homework assignments to practice the application of different machine learning algorithms in specific mechanical engineering problems and a project assignment that gives the students the flexibility of selecting their topics to study using designated machine learning tools. The overarching goal of this project is to equip mechanical engineers with machine learning skills and deepen the integration of data science into the mechanical engineering curriculum. Compared to machine learning courses offered by computer science and data science programs, this course has a much stronger focus on integration with mechanical engineering problems. Students will be provided with concrete and specific engineering problems with experimental data. The projects, presentations, and in-class peer review practice are designed to foster students' professional skills following the National Association of Colleges and Employers (NACE) competencies, including critical thinking, communication, teamwork, technology, leadership, and professionalism. Graduate students are required to complete an extra assignment (selected from three provided options) and a supercomputing assignment.

**Pre-requisites:** Graduate student standing.

**Textbook:** Steven L. Brunton, J. Nathan Kutz, Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control, 1st ed, Cambridge University Press, 2019

**Learning Outcomes:** Students completing this course are expected to be capable of

- Develop, train, and test machine learning models using Python/TensorFlow and MATLAB
- Develop machine learning models for image classification and clustering
- Perform data dimensionality reduction for physics extraction
- Analyze images/maps from experiments and simulations to predict physical quantities
- Adapt trained machine learning models to new applications
- Analyze time series for classification and regression
- Develop surrogate models for computationally expensive numerical simulations
- Benchmark the scalability of machine learning models on CPU and GPU clusters
- Develop complex machine learning models by integrating two or multiple mechanisms in tandem

**Software Packages:** Python, PyCharm, Ananconda, Google Colab, TensorFlow, Scikit-Learn, PyTorch, Keras, Numpy, Jupyter Notebook, MATLAB.

**Communication:** In this class, our official mode of communication is through uark.edu email and Blackboard. Students are responsible for checking their UARK accounts regularly. All communication between student and instructor and between student and student should be respectful and professional.

**Topics and Time Coverage:**

| Topics | Time (hours) |
|---|---|
| Introduction | 0.8 |
| Python programming and environments | 3 |
| Linear regression | 1 |
| Multilayer perceptron | 2 |
| Gaussian process regression | 2 |
| Convolutional neural networks | 3 |
| Image classification | 2 |
| Transfer learning | 2 |
| Object detection | 2 |
| Image regression | 1 |
| Fourier transform | 2 |
| Principal component analysis | 2 |
| Clustering | 2 |
| Recurrent neural networks | 2 |
| Time series analysis | 3 |
| Peer review panels | 2.8 |
| Project status updates and presentations | 6 |
| Special lecture on career and professional skills | 0.5 |
| **Total** | 39.1 |

**Grading Policy:**

| | | |
|---|---|---|
| Homework | 40% | 40 points |
| Extra assignment | 5% | 5 points |
| Supercomputing | 5% | 5 points |
| Project | 50% | 50 points |

| | | |
|---|---|---|
| Total | 100% | 100 Points |

**Grading Scale:** 90's = A, 80's = B, 70's = C, 60's = D, Below 60's = F

**Project:** Students will leverage machine learning to solve an engineering problem. Students will be provided with a few topics and can select a topic from the provided options. On the other hand, students are encouraged to think outside of the box and define the topic of their projects based on their research needs and interest. The following documents will need to be submitted for the project.

- <u>Kick-Off Presentation (2.5%):</u> Introduce the background, motivation, and scope of the project.
- <u>Status Report (2.5% × 2):</u> Update on the approach selection, model development, and preliminary results.
- <u>Quad Chart & Presentation (2.5%):</u> By the end of the semester, students will prepare a quad chart one-slider using the provided template to summarize their work and do a one-minute elevator pitch.
- <u>Final Report (40%):</u> Besides the presentation, students will also need to prepare a final report for their projects. The source codes and instructions to run the source codes should be submitted with the final report.

**Additional requirements for graduate students - Code Performance and Scaling on Supercomputers (Supercomputing):** Graduate students are required to benchmark the scaling performance of their machine learning models on CPUs and analyze the parallel efficiencies of the machine learning models.

**Additional requirements for graduate students - Extra Assignment:** Graduate students are rquired to complete at least one of the optional extra assignments. If students complete more than one extra assignments, the submission ith the highest score will be used as the final score of this assignment.

**Homework and Project Evaluation Criteria:** Students need to submit a written report for each homework assignment and each project along with their codes. The development of the machine learning models accounts for 40% of the credits, the successful implementation accounts for 30%, and the presentation/report accounts for the other 30%.

**Late Homework and Project Submission:** Students can request deadline extensions for homework assignments and projects in case of special circumstances including family emergencies, sickness, *etc*. Deadline extensions will only be offered when the student notifies the instructor ahead of time with an explanation and plan for completion. Late homework assignments and projects without prior arrangements can only account for up to 50% of the full credits.

**Student Accommodations:** University of Arkansas *<u>Academic Policy Series 1520.10</u>* requires that students with disabilities are provided reasonable accommodations to ensure their equal access to course content. If you have a documented disability and require accommodations, please contact me privately to make arrangements for necessary classroom adjustments. Please note, you must first verify your eligibility for these through the Center for Educational Access (contact ada@uark.edu or visit *http://cea.uark.edu* for more information on registration procedures).

**Academic Honesty:** As a core part of its mission, the University of Arkansas provides students with the opportunity to further their educational goals through programs of study and research in an environment that promotes freedom of inquiry and academic responsibility. Accomplishing this mission is only possible when intellectual honesty and individual integrity prevail. Each University of Arkansas student is required to be familiar with and abide by the University's 'Academic Integrity Policy' which may be found at http://provost.uark.edu/. Students with questions about how these policies apply to a particular course or assignment should immediately contact Dr. Hu. Students are not permitted to collaborate on any quiz or examination without specific permission from the instructor in advance. This includes collaboration through GroupMe, WhatsApp, or any other form of technology to exchange information associated with a quiz or examination.
The following is not all inclusive of what is considered academic misconduct for quizzes or examinations. These examples show how the use of technology can be considered academic misconduct and could result in the same penalties as cheating in a face-to-face (in person) class:

- Taking a screen shot of an online quiz or exam question, posting it to GroupMe or WhatsApp, and asking for assistance is considered academic misconduct.
- Answering an online quiz or exam question posted to GroupMe or WhatsApp is considered academic misconduct.
- Giving advice, assistance, or suggestions on how to complete a question associated with a quiz or examination is considered academic misconduct.
- The use of online websites (Quizlet, Chegg) or search engines (Google) when exam instructions indicate these are not allowed is considered academic misconduct.
- Gathering to take an online quiz or exam with others and sharing answers in the process is considered academic misconduct.

Please note: If a student or group of students are found to be exchanging material associated with a quiz or examination through any form of technology (GroupMe, WhatsApp, *etc.*) or using any unauthorized resources (Googling answers, use of websites such as Quizlet, Course Hero, Chegg, etc.), I am required to report this matter per the University of Arkansas Academic Integrity Policy. There are many websites claiming to offer study aids to students, but in using such websites, students could find themselves in violation of our University's Academic Integrity and Code of Student Life policies. These websites include (but are not limited to) Quizlet, Bartleby, Course Hero, Chegg, and Clutch Prep. The U of A does not endorse the use of these products in an unethical manner. These websites may encourage students to upload course materials, such as test questions, individual assignments, and examples of graded material. Such materials are the intellectual property of instructors, the university, or publishers and may not be distributed without prior authorization. Furthermore, paying for academic work to be completed on your behalf and submitting it for academic credit is considered 'contract cheating' per the Academic Integrity Policy. Students found responsible for this type of violation face a grading penalty of 'XF' and a minimum one-semester academic suspension per the *University of Arkansas Sanction Rubric*. Please let me know if you are uncertain about the use of a website.

**Use of Generative AI:** Students have permission to use generative artificial intelligence tools in any capacity to complete academic work in this course. Please be aware of the limitations of such tools and verify the accuracy of the content generated before submitting any work for credit.

Additionally, you are expected to properly attribute any content generated by artificial intelligence tools using the following format:

- *If AI is only used for polishing your writing:* [AI Tool/Model Name] was used to improve the readability and language of writing but was not used to generate any content.
- *If the content includes your input or editing:* This [content] (text/image/etc.) was generated by [AI Tool/Model Name] with human-guided prompts and editing.
- *If the content is completely generated by AI:* This [content] (text/image/etc.) was generated by [AI Tool/Model Name].

Please refer to the examples/guidance provided by this University of Arkansas Library Research Guide on AI and Academic Integrity for more information. The use of content generated by artificial intelligence, without proper citation, will be considered academic dishonesty and reported to the Office of Academic Initiatives and Integrity.

**Unauthorized Use of Class Recordings ante Notes:** Instructors may record class and make class available to students through Blackboard. These recordings may be used by students ONLY for the purposes of the class. Students may not download, store, copy, alter, post, share, or distribute in any manner all or any portion of the class recording, e.g. a 5-second clip of a class recording sent as a private message to one person is a violation of this provision. This provision may protect the following interests (as well as other interests not listed): faculty and university copyright; FERPA rights; and other privacy interests protected under state and/or federal law. Failure to comply with this provision will result in a referral to the *Office of Student Standards and Conduct* for potential charges under the *Code of Student Life*. In situations where the recordings are used to gain an academic advantage, it may also be considered a violation of the *University of Arkansas' academic integrity policy*.

Recording, or transmission of a recording, of all or any portion of a class is prohibited unless the recording is necessary for educational accommodation as expressly authorized and documented through the *Center for Educational Access* with proper advance notice to the instructor. Unauthorized recordings may violate federal law, state law, and university policies. Student-made recordings are subject to the same restrictions as instructor-made recordings. Failure to comply with this provision will result in a referral to the *Office of Student Standards and Conduct* for potential charges under the *Code of Student Life*. In situations where the recordings are used to gain an academic advantage, it may also be considered a violation of the *University of Arkansas' academic integrity policy*.

By attending this class, student understands the course is being recorded and consents to being recorded for official university educational purposes. Be aware that incidental recording may also occur before and after official class times.

Third parties may attempt to connect with you to buy your notes and other course information from this class. I will consider distributing course materials to a third party without my authorization a violation of my intellectual property rights and/or copyright law as well as a violation of the *University of Arkansas' academic integrity policy*. Continued enrollment in this class signifies your intent to abide by the policy. Any violation will be reported to the *Office of Academic Initiatives and Integrity*.

Please be aware that such class materials that may have already been given to such third parties may contain errors, which could affect your performance or grade. Recommendations for success in this course include coming to class on a routine basis, visiting me during my office hours, connecting with the Teaching Assistant (TA), and making use of Student Success Center. If a third party should contact you regarding such an offer, I would appreciate your bringing this to my attention. We all play a part in creating a course climate of integrity.

**Homework Assignments**

**Policies:**

- The data for homework assignments will be provided in a Box shared folder. Make sure you have access to this folder as early as possible.
- Feel free to discuss the homework assignment with your classmates. But you're prohibited from copying your classmates' codes or reports.
- If you refer to a book, paper, tutorial, website, or any other type of reference, make sure you cite the reference properly in both your reports and source codes (e.g., the authors' names, title, and links). Feel free to use open-source codes if you find them relevant and useful. Make sure to cite the reference properly in both your reports and source codes.
- Submit your homework to Blackboard by the deadline. Late submission without prior arrangements can only account for up to 50% of the full credits.
- In case of urgent conditions, you can request an extension. Your extension request will be responded to within 24 hours of submission. If your extension request is approved before the deadline of homework submission, your homework can still account for up to 100% of the full credits.
- The recommended programming languages are Python and MATLAB. But feel free to use any other languages you prefer. Note that the instructor will only be able to provide support and advice on Python and MATLAB.
- Grading Policy: The grade of your homework is majorly influenced by the breadth and depth of your practice and the details in your report. The performance metrics (accuracy, precision, loss, computation speed, etc.) of your models are not most critical, as long as you have attempted to improve the performance and reported your attempts in detail.

**Submission:**

- Homework should be submitted to Blackboard.
- Content:
  1. Report (pdf, docx, or any other format)
     - Your name
     - Instructions on how to run your code to i) train a model, and ii) test the trained model
     - Description of the methodology and results including the algorithm, model architecture, training curves, testing results, and parametric studies (different activation functions, number of layers, number of neurons, number of trainable parameters, dropout layers, etc.). Figures, schematics, and flow charts are usually very helpful for illustrations. Report training time (e.g., sec/epoch, sec till the best model) and testing time.
     - What challenges you have encountered and how you have solved them? This section is particularly important if you cannot finish the homework assignment or cannot make your code work. Describe the challenges in detail, explain what you have tried, and describe your plan to tackle the problems if you can be granted more time on this assignment.

2. Source code
3. Any other files needed for the instructor to run your code

- The zip file should be named using this format: “FirstName_LastName_HW-1.zip.”
- Do not include the data sets in the zip file.

**Homework Assignment-1: Regression**

In pool boiling experiments, the boiling heat flux can be estimated as the supplied power divided by the heater surface. However, this estimation will not be very accurate due to heat loss and other non-ideal conditions in experiments, especially for thin-film heaters with relatively low thermal conductivities (e.g., ITO heaters). Conventionally, finite-element simulations are used to evaluate the heat loss to validate or correct the experimental assumptions. Machine learning provides another perspective for tackling this issue. The heat loss and other non-ideal conditions can be captured and accounted for by the hidden layers of neural networks. The target of Problem 1-1 is to develop an MLP model to predict heat flux using temperature. The data set includes the temperature and the heat flux during a transient pool boiling test.

a. Set up and train an MLP and a GPR model to predict the heat flux based on the temperature. Report the training curves (training/validation accuracy/loss vs. epoch) and the training time (time/epoch, time till the best model).
b. Circumvent the effects of overfitting using k-fold cross-validation (e.g., using 100 foldings).

<u>Required Approaches:</u>

- Multilayer Perceptron (MLP)
- Gaussian Process Regression (GPR)

**Homework Assignment-2: Image Classification**

Pool boiling is a heat transfer mechanism that dissipates a large amount of heat with minimal temperature increase by taking the advantage of the high latent heat of the working fluids. As such, boiling has been widely implemented in the thermal management of high-power-density systems, e.g., nuclear reactors, power electronics, and jet engines, among others. The critical heat flux (CHF) condition is a practical limit of pool boiling heat transfer. When CHF is triggered, the heater surface temperature ramps up rapidly (~ 150 °C/min), leading to detrimental device failures. There is an increasing research interest in predicting CHF based on boiling images. Under the directory /ocean/projects/mch210006p/shared/HW1/classification, there are two folders, namely, “pre-CHF” and “post-CHF” that contain pool boiling images before and after CHF is triggered, respectively. The target of this problem is to develop a machine learning model to classify the boiling regime (pre or post CHF) based on boiling images.

a. Split the data set into training, validation, and testing. This can be done before training with a separate package “split-folders” or directly in the code during training.
b. Set up and train a model to classify the pre-CHF and post-CHF images. Report the training curves (training/validation accuracy/loss vs. epoch) and the training time (time/epoch, time till the best model). Use EarlyStopping for fast convergence.
c. Test the model using the reserved test data, report the confusion matrix, accuracy, precision, recall, F1 score, the receiver operating characteristic (ROC), and area under the curve (AUC).

**Homework Assignment-3: Dimensionality Reduction and Clustering**
Run dimensionality reduction and clustering analysis of the same dataset used in HW-2.

(a) Run single value decomposition (SVD) or principal component analysis (PCA) of the images and plot the percentage explained variance vs. the number of principal components (PC).
(b) Pick a representative image, run PCA and plot the reconstructed images using a different number of PCs (e.g. using PC1, PCs 1-2, PCs, 1-10, PCs 1-20, etc.).
(c) Calculate the error of the reconstructed images relative to the original image and plot the error as a function of the number of PCs.
(d) Run a clustering analysis of the boiling images using the PCs (the number of PCs to use is up to your choice) and evaluate the results of clustering.

**Reference:** Eigenface

- Wikipedia: Eigenface - Wikipedia
- Lectures by Prof. Steve Brunton (Example using MATLAB): SVD: Eigenfaces 1 [Matlab] - YouTube
- Examples using Python:
    1. EigenFaces and A Simple Face Detector with PCA/SVD in Python | sandipanweb (wordpress.com);
    2. 3.6.10.14. The eigenfaces example: chaining PCA and SVMs — Scipy lecture notes (scipy-lectures.org)
- Instructions: The PCA of a set of N images where each image has X pixels in a row and y pixels in a column should give you a total of X times Y number of components.
    1. This is achieved by converting the image (matrix of size X times Y) into a column (number of rows is X times Y, number of columns is 1) or row vector(number of rows is1, number of columns is X times Y).
    2. By doing this to N images, you will end up with a matrix (if column vector used in step (a), number of rows is X times Y, number of columns is N. If row vector used in step (a), number of rows is N, number of columns is X times Y)
    3. For example, if you have 100 images of size 1280x800, the number of PCs is 1024000. Translating this to your dataset, you should have 256x256=65536 components. So, you will get an average image of the 100 images used.
    4. Next, based on the number of PCs you want to use, the order of the dataset can be 1, 5, 10, 25, 50, or 100.

**Homework Assignment-4: Time Series Prediction**

The data file "DS-1_36W_vapor_fraction.txt" includes the vapor fraction (second column, dimensionless) vs. time (first column, unit: ms) of the boiling image sequences. The data are sampled with a frequency of 3,000 Hz (namely, a time step of 0.33 ms). Develop a recurrent neural network (RNN) model to forecast vapor fraction of future frames based on the past frames, e.g., predicting the vapor fraction profile of t = 33.33 ms – 66 ms using the vapor fraction history of t = 0.33 – 33 ms. Options include regular RNN, bidirectional RNN, gated recurrent unit (GRU), bidirectional GRU, long short-term memory (LSTM), bidirectional LSTM.

(a) Develop a baseline model with an input sequence length of 16.33 ms (50 data points) and an output sequence length of 16.33 ms (50 data points). Plot the model-predicted signal vs. the true signal.

(b) Vary the input and output sequence lengths to evaluate their effect on the error of the model predictions.

**Note:** three optional extra assignments are provided to interested students. These assignments are focused on using the machine learning algorithms practiced in Homework Assignments 1-4 in tandem to solve more complicated problems.

**Optional Extra Assignment-1: PCA-MLP in Tandem**

Re-do the image classification problem in HW-2 using PCA-MLP. Run SVD or PCA to obtain the PCs of the images. Feed the PCs to an MLP neural network to classify the regime of the boiling images.

**Optional Extra Assignment-2: Image Regression**

The vapor fraction (second) column of the data file "DS-1_36W_vapor_fraction.txt" are the labels of the images under the folder "DS-1_36W_images." Train a convolutional neural network (CNN) model to predict the vapor fraction of the images and compare the model prediction against the true data.

**Optional Extra Assignment-3: High-Dimensional Seq2Seq Learning**

The image dataset represents a boiling image sequence under transient heat loads. The images have a frame rate of 1,000 fps (or a time step of 1 ms). Run PCA to obtain the PC profiles of the image sequence. Feed the extracted PC profiles to an RNN model to forecast the PCs of future frames. Reconstruct image sequences using the predicted PCs and compare the reconstructed images against the true images. The recommended RNN models include LSTM or BiLSTM.

Note:

1. Use an input vector length of 100 and an output vector length of 100 for the model.
2. Downsample the image sequence (e.g., reducing the frame rate from 1,000 fps to 500 fps) in case of memory issues.

<u>Recommended Approaches:</u>

- Principal Component Analysis (PCA) + Long Short-Term Memory (LSTM)
- Convolutional Long Short-Term Memory (ConvLSTM)

**Sample Instructor-Defined Topic: Next-Frame Prediction of Boiling Images**

**Objective:** The target of this project is to develop a deep learning model for the accurate prediction of future boiling image sequences using past boiling image sequences (i.e., image-sequence forecasting). A variety of machine learning approaches are available for the next-frame video predictions, including convolutional neural networks (CNN), convolutional long short-term memory (ConvLSTM), autoencoder, principal component analysis (PCA), long short-term memory (LSTM), gated recurrent unit (GRU), among others.

**Key parameters metrics:**

1. Input vector length (number of frames)

2. Output vector length (number of frames)
3. The frame rate of image sequences
4. Temporal lengths of input/output vectors = input/output vector lengths × frame rate
5. The relative error between the predicted images and the true images

Parameters 1-4 are the control parameters. Parameter 5 is the evaluation metric.

**Data sets from the instructor:**

1. Boiling-20: frame rate of 1,000 fps
2. Boiling-32: frame rate of 200 fps

Both data sets are available under /ocean/projects/mch210006p/shared.

**Evaluation criteria:** The project report will be considered complete if at least one of the following perspectives is addressed.

1. Develop a deep learning model for image-sequence forecasting and explore the effect of the control parameters on the evaluation metric.
2. Implement an existing algorithm for image-sequence forecasting and revise the algorithm to demonstrate an improved performance (lower relative error between the predicted and true images).

**Quad chart template:**