CSCE 478/878 – Introduction to Machine Learning

Fall 2021

COVID-19: Policy on Face Coverings
Given current Center for Disease Control (CDC) and Lancaster county health guidance, and the current transmission level of COVID-19 in our community, I respectfully request that you join me in wearing a face covering during our classes.

Student conduct expectations: https://studentconduct.unl.edu/covid-19-related-expectations-and-non-compliance

Is this course suitable for you?

The Introduction to ML course (CSCE 478/878) is designed mainly for Computer Science and Engineering students, thus provides a rigorous theoretical treatment of the topics. It requires implementing various ML algorithms from scratch (using vanilla python and its scientific non-ML libraries). Students must have strong programming skills in Python as well as a background in probability & statistics, linear algebra, calculus, and algorithm complexity analysis. The assignments are programming-heavy and time-consuming. No support will be provided on coding or debugging for the programming assignments.

If you are a non-CS student and are only interested in how to use ML for practical applications, then this course is not suitable for you. You should instead do the "Practical Machine Learning" course, which is designed for Engineering students. It emphasizes less on the theory and mathematical foundation of the ML models as well as implementation of the ML algorithms. Also, the assignments are required to be done using ML APIs (e.g., Scikit-Learn, TensorFlow). So, a moderate level of programming skill in Python is enough. This course is offered in Spring.

Course Info

Instructor	Dr. M. R. Hasan hasan@unl.edu Avery Hall 123E https://cse.unl.edu/~hasan/
Office Hours	See Canvas
Textbook	See Canvas
Teaching Assistants	See Canvas

Prerequisite

Probability & Statistics, Linear Algebra, Calculus, Data Structure & Algorithm, Algorithm Complexity Analysis. **Strong programing skill in Python is a must**. If you don't know Python you will have to learn it independently. No help will be provided on Python programming (e.g., learning, coding, debugging).

Course Topics

This is a rough sequence of topics that we will cover in this course.

- Probabilistic Reasoning
- Linear Algebra for Machine Learning
- Classification & Regression: (Nonparametric Methods) Nearest Neighbors & K-Nearest Neighbors (K-NN)
- Classification: Performance Measures
- Kernel Density Estimator & K-NN
- Statistical Estimation of Distributions: Binomial, Bernoulli, Multinomial, Gaussian
  - Frequentist Learning (MLE)
  - Bayesian Learning (MAP)
- Regression: Linear Regression (Frequentist & Bayesian) – Closed form & Iterative Optimization Algorithm (gradient descent)
- Classification: Naïve Bayes Classifier
- Classification: Logistic Regression (Frequentist & Bayesian), Newton's Method
- Classification: Support Vector Machine; Kernel Trick
- Classification & Regression: Perceptron Algorithm & Artificial Neural Network
- Classification & Regression: Decision Tree & Random Forest
- Bagging & Boosting
- Unsupervised Learning:
  - K-means
  - Expectation Maximization (EM)
  - Gaussian Mixture Model (GMM)
- Dimensionality Reduction:
  - Principle Component Analysis (PCA)
  - Singular Value Decomposition (SVD)
  - Linear Discriminant Analysis (LDA)
- Graphical Models; Hidden Markov Model (HMM) (time permitting)
- Learning Theory (time permitting)
  - PAC
- Advanced Topics
  - Sampling Methods, MCMC, Reinforcement Learning, etc. (time permitting)

Text Resources

- Lecture slides (thorough and extensive) should provide a detailed account of the topics.

Though there is no required text for this course. However, my lectures will draw references substantially from the following three books.

- Machine Learning: A Probabilistic Perspective by Kevin P. Murphy
- Pattern Recognition and Machine Learning by Christopher M. Bishop
- Introduction to Machine learning by Ethem Alpaydin

For understanding implementation issues and hands-on insights, following text will be used:
- Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems (2nd Edition, 2019) by Aurélien Géron

Following books are useful as Python refresher/introduction:
- Data Science from Scratch by Joel Grus (O'Reilly)
- Python for Data Analysis (2nd Edition) by Wes McKinney (O'Reilley)
- Python Machine Learning by Sebastian Raschka (Packt Publishing)

Statistics, Linear Algebra & Calculus:
- Advanced Engineering Mathematics by Erwin Kreyszig
- All of Statistics: A Concise Course in Statistical Inference by Larry Wasserman
- Convex Optimization by Boyd and Vandenberghe

Discussion Through Piazza

We will be using Piazza for class discussion. The system is highly catered to provide you help fast and efficiently from classmates, TAs, and myself. Rather than emailing questions to the teaching staff, I encourage you to post your questions on Piazza. Please note that posted questions or discussion will be moderated if appeared to be inappropriate or irrelevant. Also, if you have a **question for a specific instructor (me or the GTA)**, you must send an email to that person, otherwise don't expect to receive a quick response. See the signup/login link for the Piazza on the Canvas Home page.

878 Level

There will be higher expectations with regard to the quality and quantity of work for the 878 level students compared to the 478 level students. As a student in this course, you will be expected to meet or exceed those expectations, which may require more of your time and a more proactive approach to studying and greater diligence in completing assignments. In particular, each assignment contains additional requirements or activities.

Recitations

There will be weekly recitations. A set of problems will be posted prior to the recitation. During the recitation hour the GTA will be present to help you. However, **recitations are not graded and no submission is required**. You may want to use the recitation sessions and tasks as an opportunity to deepen your understanding of the Machine Learning models that we won't have time to go over in detail during the lecture.

Homework Assignments

There will be 4/5 assignments. Assignments are due at the beginning of class on the due date. Code and other relevant files must be submitted using CSE's webhandin (http://cse.unl.edu/handin). If there are

written portions of the homework, they should be typed. You should typeset code snippets using a monotype font (Courier for example) for readability.

In this course, the homework assignments are built on prior assignments. Therefore, it is extremely important for you to stay on task and not fall behind with assignments. You should start on assignments early and resolve any and all issues (bugs, design flaws) as soon as possible so they do not carry over to the next assignment.

Coding

There will be **no support for the coding, compilation, or debugging**. You are expected to use vanilla Python and some of its scientific libraries such as Pandas, NumPy, and Matplotlib for programming assignments. However, no Machine Learning libraries such as Scikit-Learn or TensorFlow are allowed.

Exams

There will be a midterm and a final exam. See the exam schedule on Canvas Home page (Schedule section). The questions will be multiple-choice type. The exams are open book/note/computer.

Grading

Grading will be based on homework assignments, project, and quality of presentation. Note that your course total score will be computed based on the stipulated weights (see below) associated with the following graded components. **The Canvas default total score does not reflect your weighted total score.**

Assignments	70%
Midterm Exam	15%
Final Exam	15%

Scale

Letter grades will be awarded based on the following scale. I will use an **absolute grading policy**. For example, scores within the range between 89.00 and 89.49 will be rounded to 89.00 while scores within the range between 89.50 and 89.99 will be rounded to 90.00. Therefore, a score of 89.49 will earn you a B+ grade but 89.50 will earn an A-(minus) grade.

A+	>= 97	B+	>= 87	C+	>= 77	D+	>= 67	F	<60
A	>= 93	B	>=83	C	>= 73	D	>= 63		
A-	>= 90	B-	>= 80	C-	>= 70	D-	>= 60		

Grading Policy

We make every effort to grade and return materials within one academic week of the due date. You are expected to pick up your graded material in a timely manner. If you have questions about grading or believe that points were deducted unfairly, you must first address the issue with the grader it to see if it can be resolved. Such questions should be made within a reasonable amount of time after the graded

assignment has been returned. **No further consideration will be given to any assignment a week after it has been graded and returned (regardless of whether you fail to pick it up when handed back).**

It is your responsibility to **check Canvas regularly** and report missing/incorrect entry within one week of submission of your assignment or exam.

You need to report any missing/incorrect Canvas grade entry or questions related to grading **before I post your final grade** in Canvas. **I will not take any question/complaint after the final grade is posted.**

Late Work

In general, late submission will not be allowed. Exceptions may be made in certain circumstances such as health or emergency, but you must make every effort to get prior permission. Documentation may also be required.

Homework assignments have a strict *in-class* (at the beginning) due date. Any written portions should be handed in hardcopy in class while softcopies should be handed in using webhandin as specified in individual homework assignments. The webhandin program that you will use enforces a *strict* handin time based on the CSE server's clock. Programs that are even a few seconds past the due date/time will be considered late.

It is understandable that unforeseen events may interfere with your ability to submit all homework assignments on time. As such, this course allows the following late work policy: you may hand in **any *one* assignment up to 7 days late**. Any submissions after 7 days will not be considered and will be given an automatic zero. These 7 "free passes" can be used on multiple assignments.

Exceptions: You *CANNOT* use a late pass for the first assignment as well as on the last assignment.

Academic Integrity

All homework assignments, programs, and exams must be your own work unless otherwise stated. No collaboration with fellow students, past or current, is allowed unless otherwise permitted on specific assignments or problems. The Computer Science & Engineering department has an Academic Integrity Policy. All students enrolled in any computer science course are bound by this policy. You are expected to read, understand, and follow this policy. Violations will be dealt with on a case-by-case basis and may result in a failing assignment or a failing grade for the course itself. **If any student performs academic dishonesty (*complete or partial*), as defined by the UNL academic integrity policy, in any graded component of the course, her/his grade will be set to zero for that assignment/exam and she/he will be reported to the academic integrity committee of the CSE department.** The most recent version of the Academic Integrity Policy can be found at
http://cse.unl.edu/academic-integrity-policy

Communication

The best way to communicate with your Professor is through email. The Professor and teaching assistants will communicate with you either directly or through the Canvas email system. You are responsible for ensuring that the email associated with your Canvas account is up-to-date and that you are regularly

checking it. Generally speaking, you should talk to Dr. Hasan for questions about course material and lecture. You should talk to the TA about grading and homework assignments.

In addition, the Department of Computer Science & Engineering also maintains an anonymous suggestion box that you may use to voice your concerns about any problems in the course or department if you do not wish to be identified. It is available at the following URL:
http://cse.unl.edu/department/suggestion.php

Finally, I will hold regular office hours and will make myself available by appointment; please email me to set one up.

Disabilities

Students with disabilities are encouraged to contact Christy Horn for a confidential discussion of their individual needs for academic accommodation. It is the policy of the University of Nebraska-Lincoln to provide flexible and individualized accommodation to students with documented disabilities that may affect their ability to fully participate in course activities or to meet course requirements. To receive accommodation services, students must be registered with the Services for Students with Disabilities (SSD) office, 132 Canfield Administration, 472-3787 voice or TTY.