SI 670 - Applied Machine Learning

Fall 2019: Monday 5:30PM - 8:30PM 2255 NQ

Instructor: Grant Schoenebeck

Office: NQ 3411

Office hours: Monday 10:30-11:30am,
Wednesdays 2-3pm,
by appointment.

Course Assistant: Jiaqi Ma

Office hours: Tuesday 4-5pm,
Thursday 9-10am,
or by appointment.

Location: NQ1282 (Will send email notification if changed)

Contact: The course has a discussion board using Piazza set up on Canvas (sign up at: piazza.com/umich/fall2019/si670) – all questions regarding course content or that might be of interest to other students, in particular, programming or course logistics questions, should be posted to the Piazza site.

If your question requires confidentiality or is only pertinent to you, please use email.

Staff Email: SI670staff@umich.edu (use unless you need a particular instructor)

jiaqima@umich.edu (please have [SI 670 GSI] as the start of subject)

schoeneb@umich.edu

Note: syllabus details subject to change.

Course Description:

In this course, students will learn basic machine learning concepts and methods, along with how to select and apply these methods correctly to solve supervised and unsupervised machine learning problems on real-world datasets. The class focuses more on application than theory and is based on the scikit-learn library in Python. The course will start with a discussion of how machine learning is different than descriptive statistics, and introduce the scikit learn toolkit through a combination of lectures and labs. The course will introduce supervised approaches for creating predictive models, and students will be able to apply the scikit to learn predictive modelling methods while understanding process issues related to data generalizability (e.g. cross validation, overfitting). Students will also learn effective dimensionality reduction and clustering methods (unsupervised learning) to help find structure in data. The course will end with a look at more advanced techniques, such as building ensembles, deep learning, and practical limitations of predictive models. By the end of this course, students will master basic concepts of supervised (classification) and unsupervised (clustering) techniques, identify which technique they need to apply for a particular dataset and problem, engineer features to meet that need, and correctly evaluate and interpret results from their approach.

Where this course fits in the curriculum

While excellent courses in applied statistics and computer science exist that teach many of the same machine learning concepts, these courses also tend to focus more on theory rather than application, or topics that are not as immediately necessary for successful application to important real-world problems. They also include more advanced mathematical detail, and less of the practical programming focus or diversity of applications that would make this course especially relevant to SI students.

Learning Objectives

***Competency*:**

- Be able to train and apply regression and classification objects (estimators) in scikit-learn.
- Understand how to correctly prepare input data for use, e.g. feature normalization.
- Understand how to evaluate and interpret results from scikit-learn estimators.
- Understand over- and under-fitting and how to detect and prevent these.

- Use model selection methods such as cross-validation to tune the choice of model and key parameters.
- Understand how to select an appropriate machine learning method for a given scenario and dataset.

Literacy:

- Understand the tradeoffs inherent in different machine learning methods: speed, accuracy, complexity of hypothesis space, etc.
- Be able to optimize a classifier for a variety of metrics.
- Be aware of how deep learning methods work and how they are applied.

Awareness:

- What data leakage is and how to detect it.
- Issues of algorithmic bias, transparency, fairness, and other social context in machine learning applications.
- Be aware of different ensemble methods for combining classifiers.

Textbooks and Course Resources

The course will be structured in a way similar to the following recommended textbook. It's available for free online to UM students here:

Introduction to Machine Learning with Python. A. Mueller and S. Guido. O'Reilly.

http://proquest.safaribooksonline.com/book/programming/machine-learning/9781449369880/firstchapter

Off-campus: http://proquest.safaribooksonline.com.proxy.lib.umich.edu/

Additional material will be be provided in lectures, readings, and other handouts.

The Deep Learning section of the course will be using
Deep Learning with Python, by Francois Chollet. Manning.

It's also available for free to UM students here:
https://proquest.safaribooksonline.com/book/programming/python/9781617294433

Another resource is the Applied Machine Learning Coursera course (Course 3, available for free) as part of the UM Applied Data Science in Python Specialization - which provides videos, background material and examples and can be accessed for free here: https://online.umich.edu/catalog/69/

If you need help getting up to speed on pandas and numpy, you can take Course 1: Introduction to Data Science in Python (also for free)
https://online.umich.edu/catalog/50/

Class Format

The course will combine lectures with in-class lab sessions. **You'll need to bring your laptop to each lab: please let us know *right away* if that's not possible.** Students learn the concepts and techniques in the lecture and practice them by writing programs in the lab portion of the class. In addition to the labs, there will be regular programming assignments and a midterm. Students will use Python with Jupyter / Colab for most of the labs and homework assignments. A key part of the class will be a final project.

Course Schedule

Note: Timing almost certainly will change.

Data	**Topics**
Sept 9	Week 1 Course Introduction; Data exploration (review), Machine Learning Process, K-NN, feature normalization
Sept 16	Week 2 Basic supervised learning concepts.

	Linear regression, feature expansion, regularization: lasso, ridge, measuring error: RMS error, k-fold cross validation. Overfitting and underfitting
Sept 23	Week 3 Logistic regression. Measuring accuracy: ROC, confusion matrix, dealing with categorical and missing data.
Sept 30	Week 4 Support vector machines (linear and kernelized): RBF kernels, feature importance, Multi-class classification. Gradient Descent and EM. Boot-strapping. Hyper-parameter search
Oct 7	Week 5 Decision trees for classification and regression, entropy Boosting, Random forests, gradient boosted decision trees, XGBboost Data Leakage
Oct 14	Week 6 No class - Fall Break
Oct 21	Week 7 Bayesian Classifiers, Probability, Unsupervised learning: density estimation.
Oct 28	Week 8 Midterm Examination Unsupervised learning: clustering. Agglomerative/tree-based clustering. K-means and variants. EM algorithm.
Nov 4	Week 9 Unsupervised learning: dimensionality reduction (PCA, multi-dimensional scaling, t-SNE) Evaluation of unsupervised methods.

Nov 11	Week 10: Deep learning 1.Neural networks, Convolutional NN, Embeddings.
Nov 18	Week 11: Deep learning 2. Visualizing ConvNets. Sequence problems: Recurrent NN.
Nov 25	Week 12: Deep learning 3 Generative Adversarial networks (GANs)
Dec 2	Week 13: FAT-ML: bias in training and data collection; implications of privacy
Dec 9	Week 14: Final project presentations (or catch-up). Incentives and Learning, adversarial ML

Class Format

The first half of the class will be typically be lecture-oriented or with interactive activities (e.g. collaborative break-outs). From time to time there may be project-related presentations or activities. The second half will typically be lab-oriented, with interactive problem-solving sessions that are meant to give you an opportunity to get experience actually writing code that uses whatever new tools and concepts have been covered in class that week.

Typically, the instructor and/or GSIs will make slides available online via Canvas before the class.

Homeworks

Homeworks will be due by 11:59pm on the date given in the syllabus and designated in Canvas. Please see the late policy section for what happens if you can't make this deadline.

Please make sure you read the section below on Giving/Receiving Help. If you copy someone else's homework solution completely, or almost completely (and/or failing to acknowledge your source), then this will be considered cheating, and I'll refer your case to the academic advising office for disciplinary action.

Please contact the instructor if you have any uncertainty or questions about this policy.

Attendance

Attendance at lectures and participation in labs is necessary for getting the most out of the course.

Readings

There will be readings assigned several weeks that are meant to give you important preparation for the lecture and/or lab. You're highly encouraged to read these beforehand, since you're likely to get much more out of the class or lab if you do.

Electronics Policy

Much research and experience suggests that electronics are distracting during class and hurt everyone's ability to learn, so during any lectures or other non-lab presentations that don't require you to actually be writing code, please do not use laptops, phones, tablets, etc. unless you have a specific class-related need to do so.

Giving and Receiving Assistance

I'm going to state a policy used in other SI programming-based courses (e.g. SI106 and SI370 (the following text is taken in modified form from the SI370 F15 syllabus).

Learning technical material is often challenging, and a course like this one covers a range of topics and can move quickly. We want you to succeed in the course and we encourage you to get help from anyone you like.

However: In the end, **you** are responsible for learning the material – so you need to make sure that the help you get is focused on gaining knowledge, not just on getting through the assignments. If you rely too much on help so that you fail to master the material, especially the basics earlier in the semester, you will crash and burn later in the course.

The final submission of each homework exercise must be in your own words. If you get help on an assignment, please indicate the nature and amount of help you received. If the assignment involves computer code, add a comment indicating who helped you and how. Any excerpts from the work of others must be clearly identified as such (e.g. quotation with citation, or with comments in the code if it is a code fragment you have borrowed).

If you're a more advanced student and are willing to help other students, please feel free to do so. Just remember that your goal is to help teach the material to the student receiving the help. It is acceptable for this class to ask for and provide help on an assignment via the Q&A site (Piazza), including posting short code fragments (e.g. 2-3 lines). Just don't post complete answers. If it seems like you've posted too much, one of the instructional staff will contact you to let you know, so don't worry about it. When in doubt, err on the side of helping your fellow students. To reiterate, the collaboration policy is as follows. Collaboration in the class is allowed (and even encouraged) for assignments – you can get help from anyone as long as it is clearly acknowledged. Collaboration or outside help is also not allowed on exams or other types of assessments, though for major tests you will be allowed to bring in specific "cheat sheets" with you. Use of solutions from previous semesters is not allowed. The authorship of any assignments must be in your own style.

Final Project

The goal of the final project is to further apply what you've learned in class to real-world datasets. The deliverables for the final project, which will be due at the end of the course, will include an initial proposal, poster presentation, final report, and code/data repository. Details on the final project will be available as we get closer to the project proposal date. You may work individually or in groups of 2-3. However, the scale of the project should reflect the number of participants.

Grading

The graded work in the course will be weighted as follows to determine a final percentage grade:

Homeworks: 55%

Midterm: 15%

Final project: 30%

Conversion between percentage grades and letter grades will use the following mapping:

A+ 98; A 95; A- 90; B+ 87; B 83; B- 80; C+ 77; C 73; C- 70; D+ 67; D 63; D- 60; E 50; F 40.

Bonus Credits

There may be opportunities for bonus credits during the course.

Late Policy

A **25% penalty** will be assessed for each subsequent 24h period after the deadline that an assignment is late up to 48 hours, after which the assignment will not be accepted. For example, if the due date is 11:59pm on Thursday, the possible late penalties would be:

Before 11:59pm Friday: 25% deduction

Before 11:59pm Saturday: 50% deduction

After 11:59pm Saturday: 100% deduction

The deduction will be taken from the maximum allowable points. For example, if you score 90/100 and turn the assignment in 35 minutes late, you will receive a score of 65/100.

We realize that the occasional crisis might screw up your schedule enough to require a bit of extra time in completing a course assignment. Thus, we have instituted the following late policy that gives you a limited number of flexible “late day” credits. These late days are intended to be used in exceptional/emergency circumstances (illness, computer died, power outage), so you should have plenty to get through the term.

You have **four (4)** free late days to use during the course. No fractional late days: they are all or nothing. Each late day used will void one 25% penalty. For example, if an assignment is 3 days late, and you use 2 late days, then it will be assessed a 25% penalty. **No assignment can be turned in more than 2 days late**. Therefore, you cannot use late days on an assignment that is 2 or more days late.

You don't need to explain or get permission to use late days, and we will track them for you. In cases where late days can be assigned in multiple ways (e.g. you have only one late day left but hand in two late assignments) we will always allocate late days in a way that maximizes your grade. Note that resubmissions after the deadline will be counted as late submissions. **Also, late days may not be applied to the final project.**

Exams

There is a midterm exam in this course.

Classroom policy

Students are asked to attend class on time and remain through the entire class. Students will need to bring their laptops for the in-class labs.

As described above, I won't take attendance in every class, but may check it from time to time and factor this into the course participation part of your overall grade.

Academic Integrity and Misconduct

Collaboration
UMSI strongly encourages collaboration while working on some assignments, such as homework problems and interpreting reading assignments as a general practice. Active learning is effective. Collaboration with other students in the course will be especially valuable in summarizing the reading materials and picking out the key concepts. You must, however, write your homework submission on your own, in your own words, before turning it in. If you worked with someone on the homework before writing it, you must list any and all collaborators on your written submission. Each course and each instructor may place restrictions on collaboration for any or all assignments. Read the instructions careful and request clarification about collaboration when in doubt. Collaboration is almost always forbidden for take-home and in class exams.

Plagiarism
All written submissions must be your own, original work. Original work for narrative questions is not mere paraphrasing of someone else's completed answer: you must not share written answers with each other at all. At most, you should be working from notes you took while participating in a study session. Largely duplicate copies of the same assignment will receive an equal division of the total point score from the one piece of work.

You may incorporate selected excerpts, statements or phrases from publications by other authors, but they must be clearly marked as quotations and must be attributed. If you build on the ideas of prior authors, you must cite their work. You may obtain copy editing assistance, and you may discuss your ideas with others, but all substantive writing and ideas must be your own, or be explicitly attributed to another. See the (Doctoral, MSI, MHI, or BSI) student handbooks available on the UMSI intranet for the definition of plagiarism, resources to help you avoid it, and the consequences for intentional or unintentional plagiarism.

More specifically for programming assignments and data science projects, unless explicitly specified, <u>all submitted work must be your own, original work</u>. As with other forms of writing, you may discuss <u>general</u> approaches with others on individual assignments, but you should work on the code by yourself. It is a violation of the original work policy to copy code or other work wholesale. If you did work closely with any other students that helped you with any assignment, you must indicate on your turned-in assignment who you worked with, and how.

If you are taking or have taken another data science-related course this term you may not submit the same work to more than one course (including from a previous course) for your final project. While you might work on a broadly similar problem area, the code and report you turn in must be substantially different and must be unique to this course. Please see the instructor if you're not sure what this means.

Accommodations for Students with Disabilities

If you think you need an accommodation for a disability, please let me know at your earliest convenience. Some aspects of this course, the assignments, the in-class activities, and the way the course is usually taught may be modified to facilitate your participation and progress. As soon as you make me aware of your needs, we can work with the Services for Students with Disabilities (SSD) office to help us determine appropriate academic accommodations. SSD (734-763-3000; http://ssd.umich.edu) typically recommends accommodations through a Verified Individualized Services and Accommodations (VISA) form. Any information you provide is private and confidential and will be treated as such.

Student Mental Health and Wellbeing

The University of Michigan is committed to advancing the mental health and wellbeing of its students, while acknowledging that a variety of issues, such as strained relationships, increased anxiety, alcohol/drug problems, and depression, directly impacts students' academic performance. If you or someone you know is feeling overwhelmed, depressed, and/or in need of support, services are available. For help, contact Counseling and Psychological Services (CAPS) at (734) 764-8312 and https://caps.umich.edu/ during and after hours, on weekends and holidays or through its counselors physically located in schools on both North and Central Campus. You may also consult University Health Service (UHS) at (732) 764-8320 and https://www.uhs.umich.edu/mentalhealthsvcs, or for alcohol or drug concerns, see www.uhs.umich.edu/aodresources. For a more comprehensive listing of the broad range of mental health services available on campus, please visit: http://umich.edu/~mhealth/.