CUSP-GX 5006.001: Machine Learning for Cities

Spring 2019
Lecture: Thursdays 9:30 am – 12:20 pm
CUSP, 2 MTC, 9th Floor, Room 9.009

Instructor:
Professor Daniel Neill, daniel.neill@nyu.edu

Course Assistant:
Harshit Srivastava, hs3500@nyu.edu

Office Hours:
Professor Daniel Neill (daniel.neill@nyu.edu)
Tuesdays 5:00-5:45pm and Thursdays, 1:30-2:15pm, **starting 2/7**. Office: 370 Jay Street #1301D.

[Please feel free to stop by without an appointment during my office hours; meetings at other times are by appointment only.]

Harshit Srivastava (hs3500@nyu.edu)
Mondays and Fridays, 3:00-4:00pm, Prague Conference Room (370 Jay Street #1329).

Course Description and Objectives

The objective of this course is to familiarize students with modern machine learning techniques and demonstrate how they can be effectively applied to urban data. The course is practice-oriented: concepts and techniques are motivated and illustrated by applications to urban problems and datasets. For that reason, it involves a significant programming component, with Python as the primary programming language. Topics include a variety of supervised and unsupervised learning methods, such as support vector machines, clustering algorithms, ensemble learning, Bayesian networks, Gaussian processes, and anomaly detection. Strategies for effective machine learning and discussion of the limitations of ML as well as a variety of supplementary techniques are also considered.

The primary objectives of the course are to enable students to:

1) Understand the motivation behind different machine learning methods and their applicability in a given practical context.
2) Be able to implement methods adapted to the problem at hand using existing software libraries.
3) Know how to interpret the results appropriately.

Necessary background theory is provided; however, the course does not focus on theory. That said, students who wish to engage more with the theory behind machine learning methods are encouraged and supported.

Course Requirements

Principles of Urban Informatics I (CUSP-GX 5003) and Applied Data Science (CUSP-GX 5004) or equivalent. Alternatively, undergraduate coursework in computer science and statistics or some equivalent experience (including basic proficiency in Python) could be sufficient for following the course. Students who are unsure if they satisfy the prerequisites should check with the course instructor.

Course Structure
Class sessions will typically include ~1.5 hours of slide-based lecture presentations, followed by a ~1 hour interactive coding session.

Grading:
1) Midterm exam (20%) – March 28th
2) Four homework assignments (each 10% of the grade, for a total of 40%) – due 2/28, 3/14, 4/18, 5/2.
3) Final project report (30%) – due May 9th.
4) Final project presentation (10%) – date TBD during exam week (during final exam slot or regular class time)

Readings:
This course will primarily be based on the instructor's course notes, supplemented by articles or book chapters when necessary. There is no required textbook. Recommended *reference* textbooks include:

1) "The Elements of Statistical Learning" by Hastie, Tibshirani, and Friedman (available for free download at http://statweb.stanford.edu/~tibs/ElemStatLearn/)
2) "Pattern Recognition and Machine Learning" by Christopher M. Bishop
3) "Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy
4) "Introduction to Machine Learning, Second Edition" by Ethem Alpaydin (http://cs.du.edu/~mitchell/mario_books/Introduction_to_Machine_Learning_-_2e_-_Ethem_Alpaydin.pdf)
5) "Machine Learning" by Tom Mitchell
6) "Data Science for Business" by F. Provost and T. Fawcett

Software:
This course will use a variety of software tools and packages. Python (usually through ipython notebooks including packages like pandas, numpy and sklearn) will be the primary programming language. There will be a significant programming component and basic exploratory, modeling, and visualization abilities will be assumed.

Late Work Policy:
You are expected to turn in all work on time (at the start of class on the due date). Because we understand that exceptional circumstances may arise, each student will be permitted to turn in one assignment up to 48 hours late with no penalty. Any other late assignments will not be accepted.

NOTE: Assignments turned in more than **five (5)** minutes after class starts will be counted as "late" and treated according to the Late Work Policy above.

Course Schedule (tentative and subject to change):

Date	Lecture	Topics
2/7	01	**Introduction to Machine Learning for Cities** Course structure and syllabus Course overview- methods and example applications Supervised and unsupervised learning Common ML problem paradigms- classification, regression, clustering, modeling, detection Python review part 1: basics. **[NOTE: THIS IS THE FIRST CLASS MEETING. NO CLASS AND NO OFFICE HOURS THE WEEK OF 1/28-2/1.]**
2/14	02	**Interpretable Classification (and Regression) using Decision Trees** The prediction problem (classification and regression) Overview of classification methods; discussion of interpretable vs. black-box predictors Interpretable prediction: Rule-based, instance-based, and model-based Rule-based Learning: Decision trees for classification and regression Python review part 2: data science with Python.
2/21	03	**Ensemble Methods and the Accuracy vs. Interpretability Tradeoff** Overview of ensemble methods (stacking, boosting, bagging, random forests) From trees to forests: detailed discussion of random forests Revisiting accuracy vs. interpretability Decision trees and random forests in Python
2/28	04	**From Linear to Non-Linear Classifiers with Support Vector Machines** ***** Homework Assignment 1 (Trees & Forests) due at the start of class ***** Linear decision boundaries. Support vector machines (SVMs) for classification. Why maximize the margin? Moving from linear to non-linear decision boundaries with kernel SVM. SVM in Python.
3/7	05	**Bayesian Methods for Supervised, Unsupervised, and Semi-Supervised Learning** Comparison of supervised, unsupervised, and semi-supervised learning Applications of clustering for modeling group structure Naïve Bayes (NB) classification. A simple supervised learning algorithm. Naive Bayes vs. logistic regression. Expectation-maximization (EM) for clustering and semi-supervised NB classification. NB and EM in Python.
3/14	06	**More clustering algorithms** ***** Homework Assignment 2 (SVMs, Naïve Bayes, EM) due at the start of class ***** Brief digression: clustering as a search problem Hierarchical clustering: bottom-up and top-down K-means clustering Leader clustering for massive streaming data Clustering in Python.

		****Happy Spring Break! (no class on 3/21 and no office hours that week) ****
3/28	07	**Midterm Exam [CLASS ENDS AT 11AM- NO LAB TODAY]**
4/4	08	**Bayesian Networks** Bayes Nets: a tool for modeling the relationships between multiple variables Other uses of Bayes Nets (such as classification and anomaly detection) Efficient representation with Bayes Nets Building and interpreting Bayes Nets Inference of conditional dependencies with Bayes Nets Learning Bayes Net Parameters from Data Learning Bayes Net Structure from Data Bayes Nets with Python
4/11	09	**Causal Structure Learning with Bayesian Networks** Causal structure learning with the PC algorithm Assumptions and comparison to econometric methods Extensions and variants Causal Orientation methods
4/18	10	**Gaussian Processes** ****Homework Assignment 3 (Clustering & Bayes Nets) due at the start of class **** Modeling spatial and temporal dependencies and other non-iid data GP regression Scaling up GP regression; long range forecasting Other uses of GPs: causal inference and change point detection
4/25	11	**Anomaly and Pattern Detection Part 1** Model-based anomaly detection Detecting anomalies using Bayesian networks Distance-based, cluster-based, and density-based anomaly detection
5/2	12	**Anomaly and Pattern Detection Part 2** ***** Homework Assignment 4 (GPs & Anomaly Detection) due at the start of class ***** Detecting patterns of anomalies Spatial cluster detection and event detection Applications to disease surveillance
5/9	13	**Fairness, Accountability, and Transparency in Machine Learning** ***** Final Project Reports due at the start of class ***** Definitions of fairness and bias Fairness-aware algorithms Automatically detecting biases in classification **[NO LAB TODAY]**
5/16??		**Final Project Presentations during exam week (no final exam)**

Acknowledgements
Some of the material for this course has been borrowed from earlier iterations, and is used with permission. Thanks very much to Drs. Stanislav Sobolevsky, Martin Jankowiak, and Ravi Shroff for their contributions to the course content.

Statement of Academic Integrity

NYU-CUSP values both open inquiry and academic integrity. Full and Part-Time graduate programs and advanced certificate programs are expected to follow standards of excellence set forth by New York University. Such standards include but are not limited to: respect, honesty and responsibility. The program has zero tolerance for violations to academic integrity. Such violations are deemed unacceptable at NYU and CUSP. Instances of academic misconduct include but are not limited to:

- Plagiarism
- Cheating
- Submitting your own work toward requirements in more than one course without
    a) Prior documented approval from instructor and
    b) Proper citation
- Forgery of academic documents with the intent to defraud
- Deliberate destruction, theft, or unauthorized use of laboratory data, research materials, computer resources, or University property
- Disruption of an academic event (lecture, laboratory, seminar, session) and interference with access to classroom, laboratories, or academic offices or programs

Students are expected to familiarize themselves with the University's policy on academic integrity and CUSP's policies on plagiarism as they will be expected to adhere to such policies at all times – as a student and an alumni of New York University